.Mac SDK: What's New in 2.0

What's new for Mac OS X "Tiger"?

Session 119

These are confidential sessions—please refrain from filming or taking pictures.

Application Technologies

.Mac SDK: What's New in 2.0

What's new for Mac OS X "Tiger"?

Scott Ryder
Director Engineering, .Mac

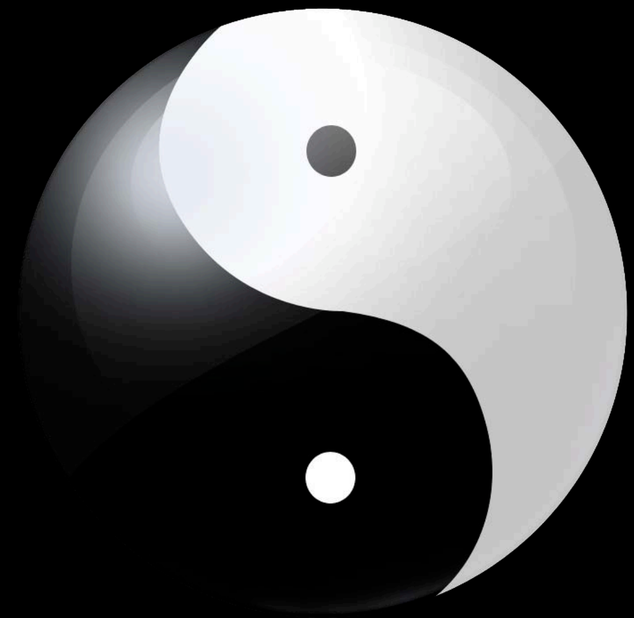
http://developer.apple.com/wwdc2005/

Who Can Benefit From the Kit?

- Games
  - Multi-player games
  - Games that have community
  - Turn based games
- Document-centric applications
  - Shared documents
  - Data backup or secure distribution
  - Workflow/collaboration
- Applications that require reliable notification
  - Distributed solutions
  - Resource tracking/reservation
- Inter-application communication
  - Suites

Agenda

- Overview of .Mac for Developers
- .Mac Kit 1.0
  - What is it?
  - What are developers doing with it?
- Introducing .Mac Kit 2.0
  - Feature overview
  - What can you build?
  - Demos
  - API review
- Q&A



.Mac Overview for Developers

The Tao of .Mac Integration

http://developer.apple.com/wwdc2005/

Why Integrate with .Mac?

- Key features
  - Global namespaces
  - Integrated with Mac OS X
  - Seamless server to client
- Enables new capabilities
  - Durable identification of users
  - Publish to web
  - Store and forward
  - Collaboration
  - Security and access control

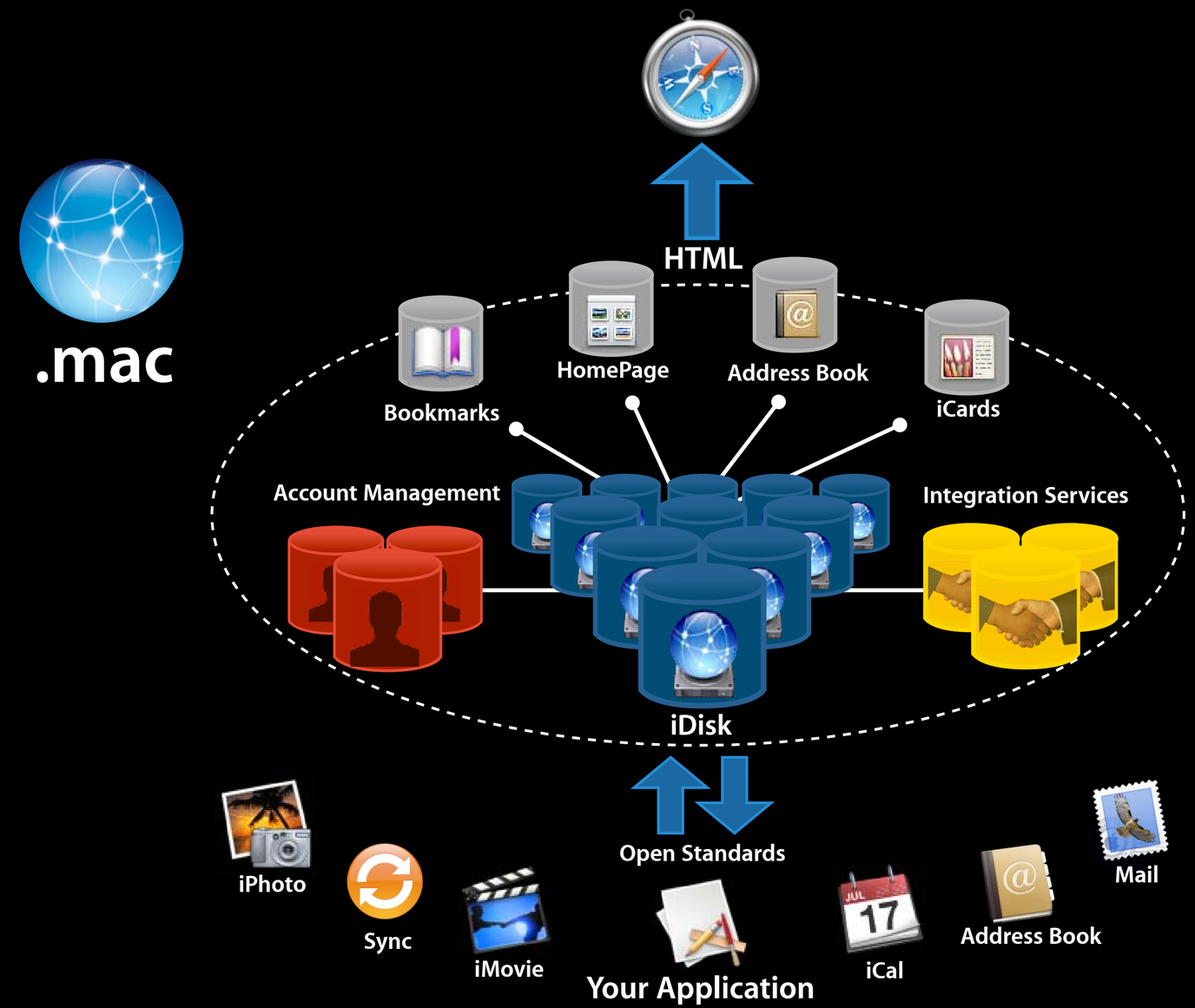
What Does .Mac Offer?

- At no cost to your application .Mac provides
  - Access to well over ½ million paying members
  - Account management
  - Scalable services
    - 10+ TB downloaded and 1½+ TB uploaded per week
    - Processes ½ billion transactions per week
  - Systems support and monitoring 24/7
  - Open standards based

- You earn $15 for every member you sign up

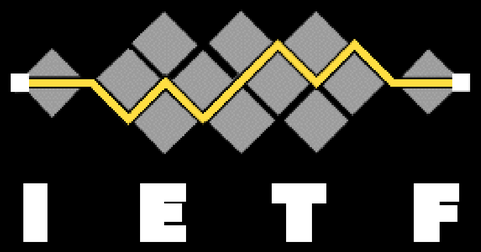
.Mac: A Developer's Perspective

High level perspective on .Mac architecture



iDisk Is the Core

- All user data lives on iDisk
  - .Mac Application Servers also use iDisk
  - Complete support for WebDAV Level 2 (RFC #2518)
- New for 2005
  - Full UTF-8 support
  - Complete support for WebDAV Access Control Lists (RFC #3744)
    - Very flexible ACL Model
    - User definable groups
    - Multiple credentials support
      - All are first class citizens in terms of Access Control Lists



On Standards…

- Central to our philosophy
  - Committed to standards that are
    - Stable
    - Secure
    - Broadly accepted
    - Offer distinct user benefits
- Today we support
  - HTTP, HTTPS, WebDAV Level 2, and WebDAV ACL
  - IMAP, POP, and SMTP
  - XML RPC
  - Others…

Overview of .Mac Kit 1.0

The easiest way to integrate with .Mac

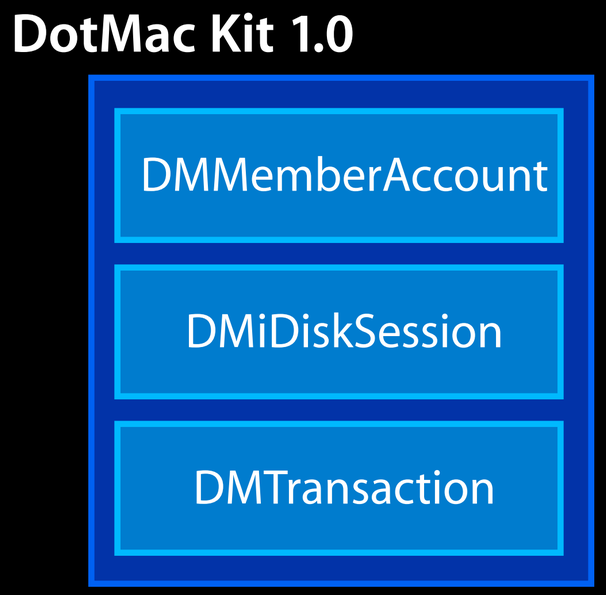
http://developer.apple.com/wwdc2005/

What Is the .Mac Kit?

- .Mac is Open Standards based but…
  - Using these standards requires complex network programming
  - Why do all that work if you don’t have to?
- .Mac Kit (DotMac Kit)
  - Built on high performance Mac OS X infrastructure
  - Simple session style Objective-C Cocoa API
  - Fully supports both sync and async operations
  - Embedded Framework

DotMac Kit 1.0

- Three core classes
  - DMMemberAccount
    - Deals with member accounts
  - DMiDiskSession
    - Manages access to iDisk
  - DMTransaction
    - Encapsulates all interactions with .Mac
    - Heart of the kit
- Fully supports sync and async
- Thread-safe



Using DMTransaction

- Determining progress…

```
SInt64 bytesToMove = [inflightTransaction contentLength];
SInt64 bytesMovedSoFar = [inflightTransaction bytesTransferred];
```

Using DMTransaction

- Determining progress…

```
SInt64 bytesToMove = [inflightTransaction contentLength];
SInt64 bytesMovedSoFar = [inflightTransaction bytesTransferred];
```

- Checking state…

```
int transactionState = [inflightTransaction transactionState];
switch (transactionState) {
  case kDMTransactionNotStarted:
  case kDMTransactionSuccessful:
  …
}
```

Using DMTransaction

- Determining progress…

```
SInt64 bytesToMove = [inflightTransaction contentLength];
SInt64 bytesMovedSoFar = [inflightTransaction bytesTransferred];
```

- Checking state…

```
int transactionState = [inflightTransaction transactionState];
switch (transactionState) {
  case kDMTransactionNotStarted:
  case kDMTransactionSuccessful:
  …
}
```

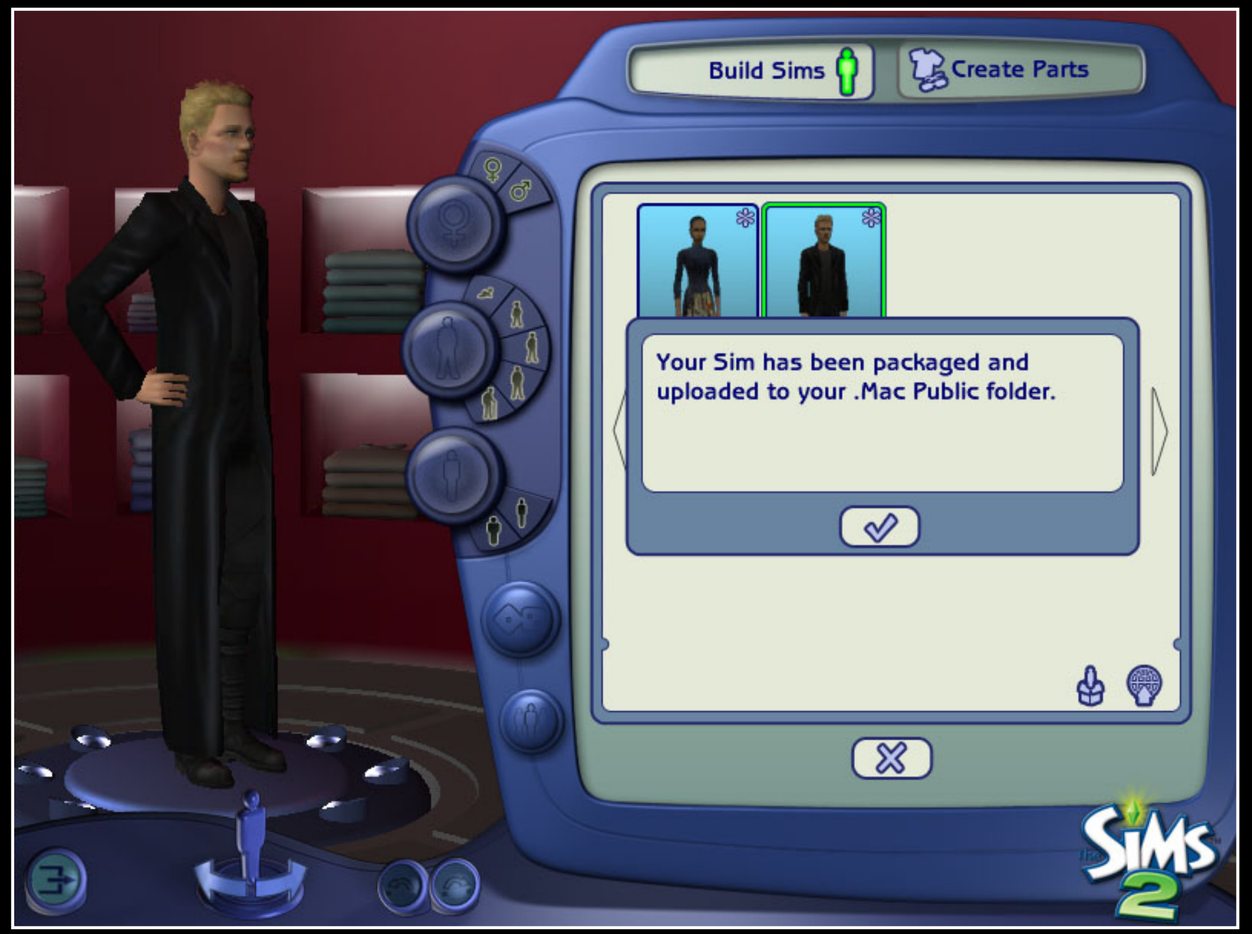
- Getting the result…

```
if ([inflightTransaction isSuccessful]) {
  result = [inflightTransaction result];
}
```

Who Is Using the Kit Today…

- .Mac Application Servers
- Used extensively by Apple
  - Mac OS X "Tiger"
  - iLife
  - .Mac applications like backup
- Over 10,000 downloads of the .Mac kit
  - Over 10 million transactions/week
    - Web publishing
    - Backup of data files
    - Collaboration…

A Fun Thing 1.0 Enables



So where do we go from here?

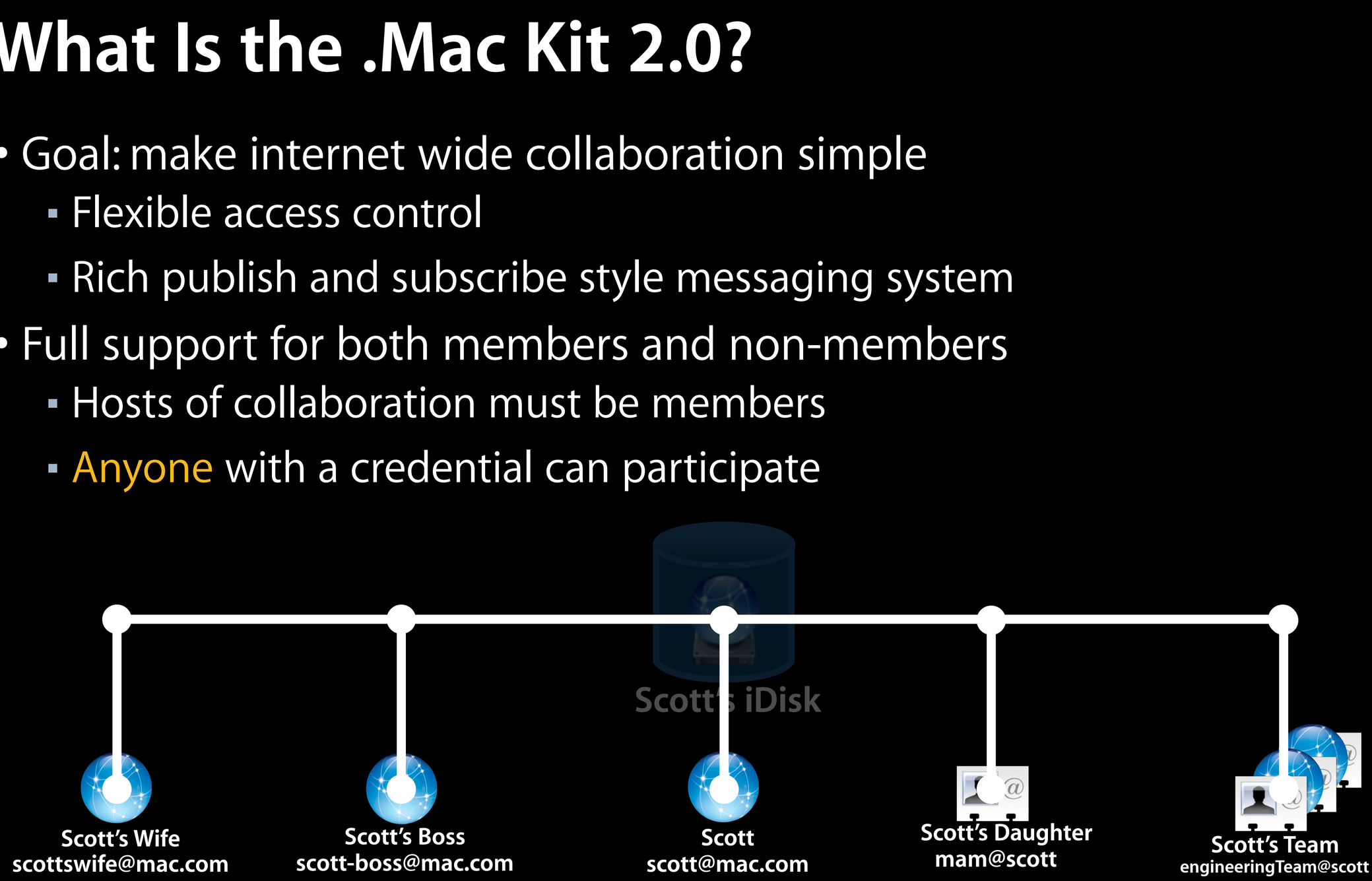
.Mac Kit 2.0

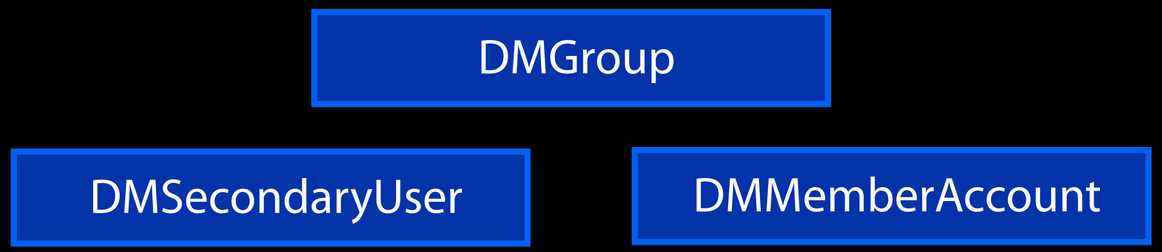
The easiest way to add community and collaboration to your Application

http://developer.apple.com/wwdc2005/

What Is the .Mac Kit 2.0?

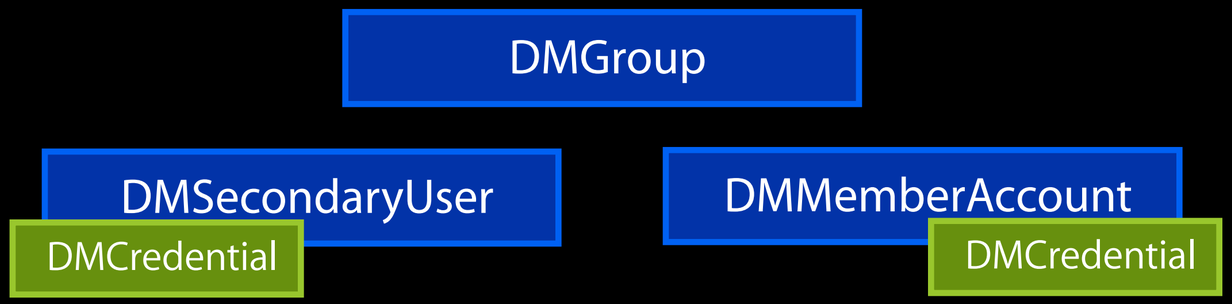
- Goal: make internet wide collaboration simple
  - Flexible access control
  - Rich publish and subscribe style messaging system
- Full support for both members and non-members
  - Hosts of collaboration must be members
  - Anyone with a credential can participate





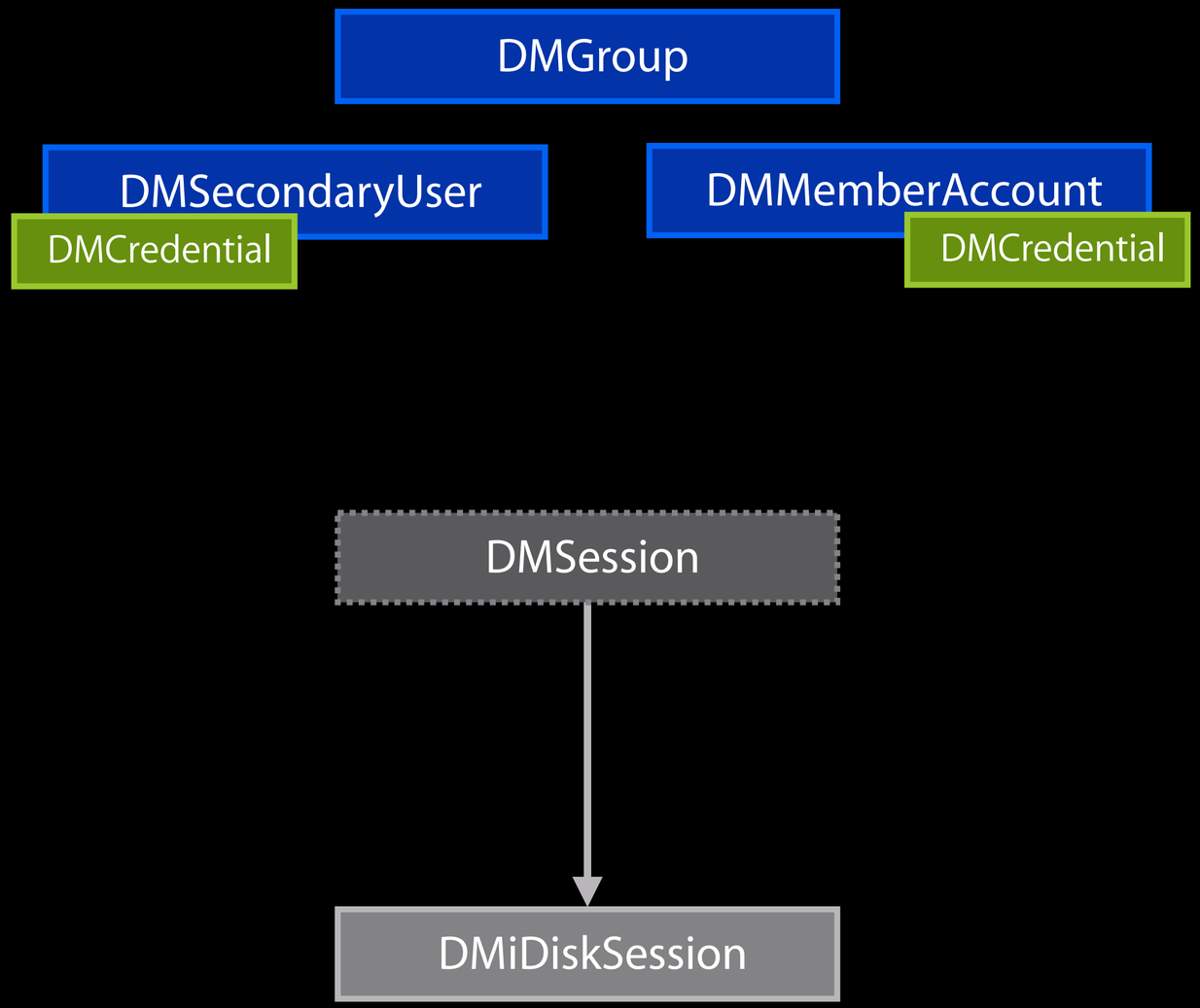
DotMac Kit 2.0

- New “Credentials” features
  - DMSecondaryUser (Class)
  - DMGroup (Class)



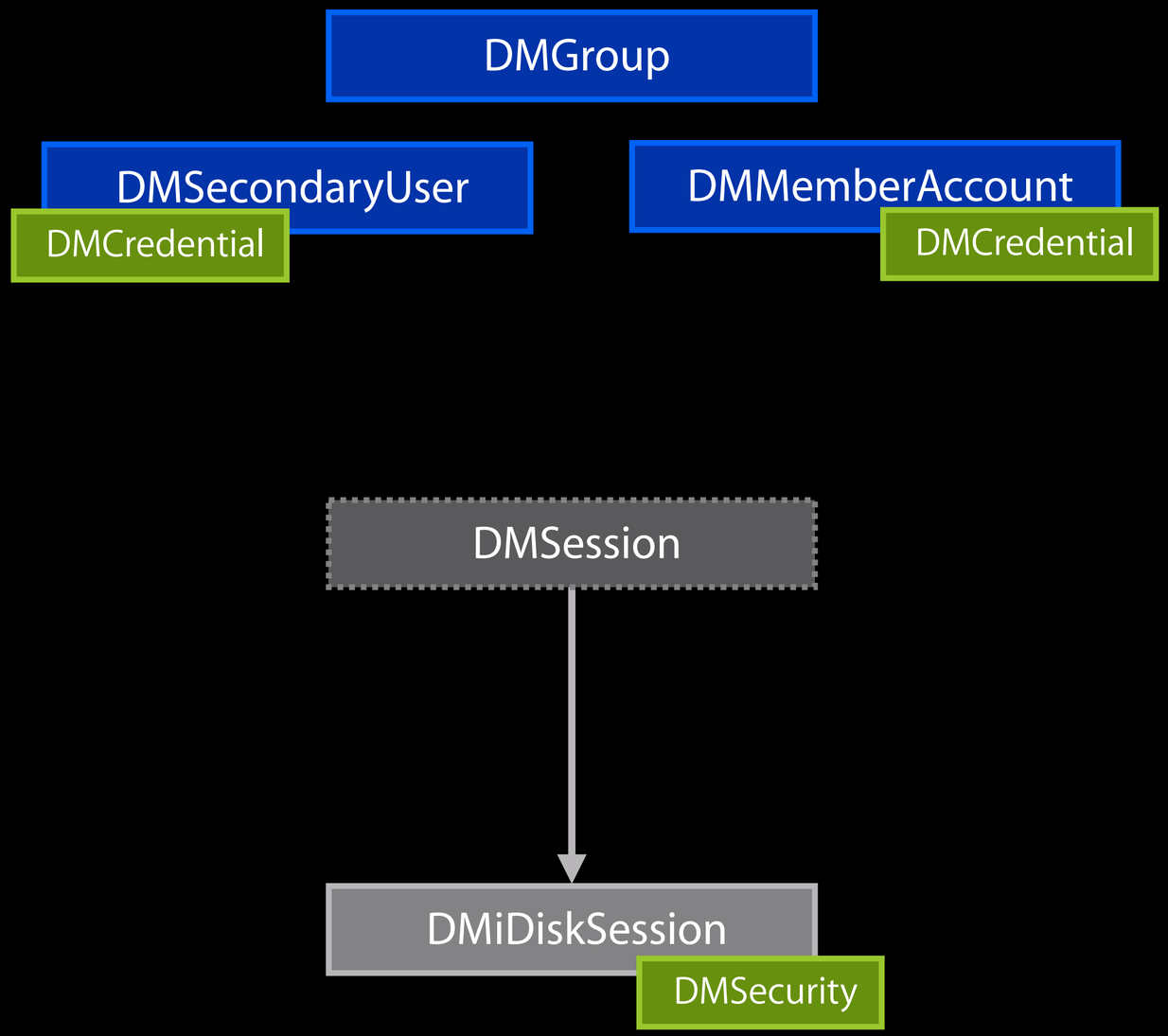
DotMac Kit 2.0

- New “Credentials” features
  - DMSecondaryUser (Class)
  - DMGroup (Class)
  - DMCredential (Formal Protocol)



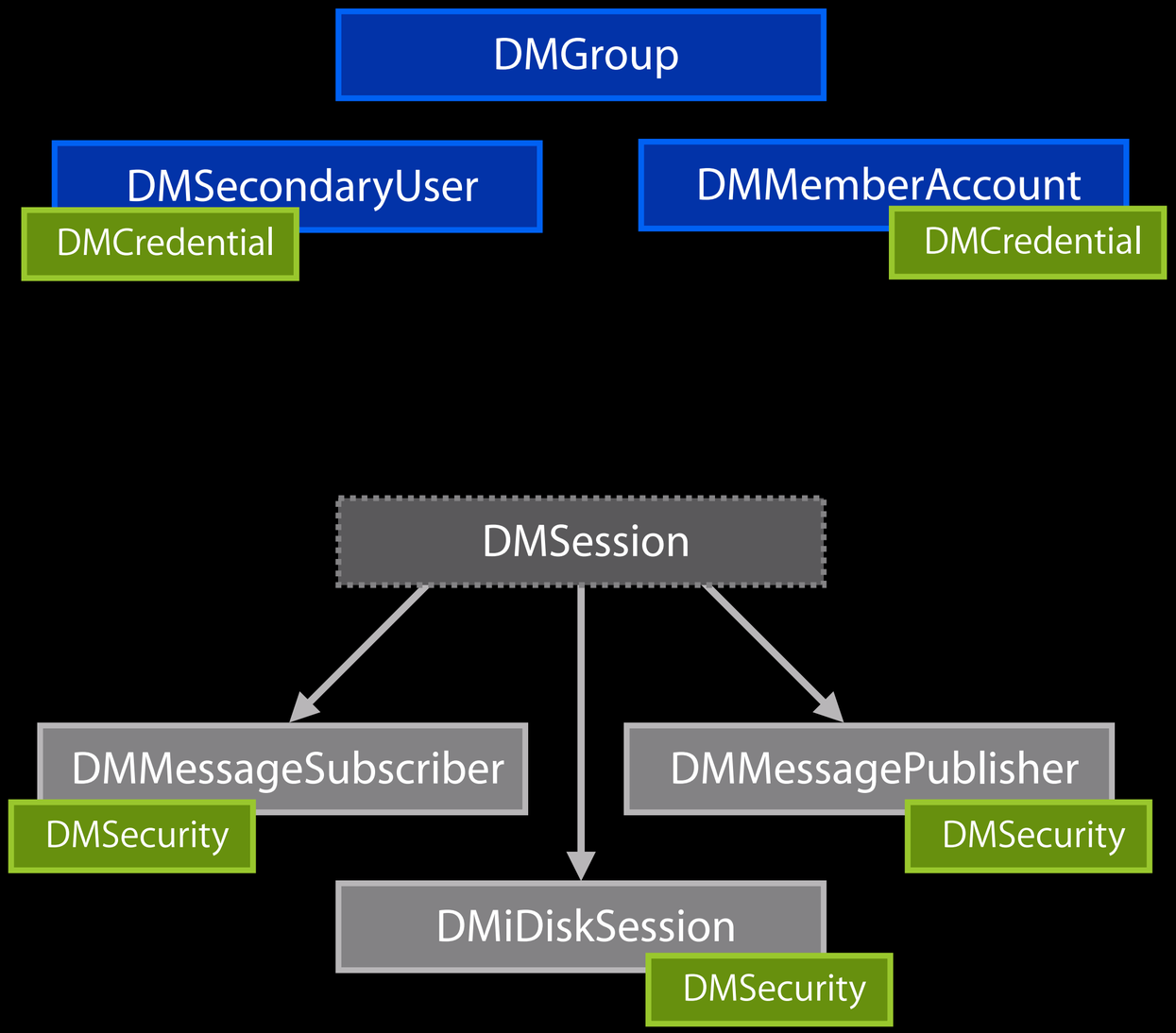
DotMac Kit 2.0

- New "Credentials" features
  - DMSecondaryUser (Class)
  - DMGroup (Class)
  - DMCredential (Formal Protocol)
- New abstract Base-Class for services
  - DMSession



DotMac Kit 2.0

- New "Credentials" features
  - DMSecondaryUser (Class)
  - DMGroup (Class)
  - DMCredential (Formal Protocol)
- New abstract Base-Class for services
  - DMSession
- New security features
  - DMSecurity (Formal Protocol)

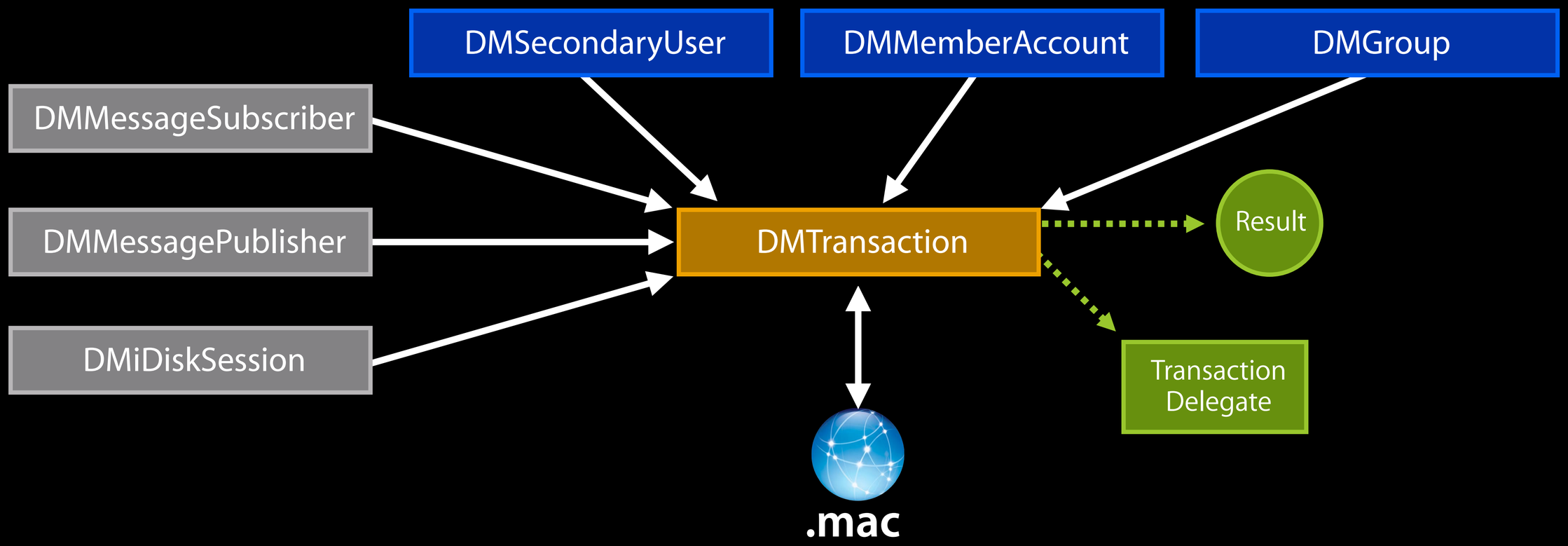


DotMac Kit 2.0

- New Identity features
  - DMSecondaryUser (Class)
  - DMGroup (Class)
  - DMCredential (Formal Protocol)
- New abstract Base-Class for services
  - DMSession
- New security features
  - DMSecurity (Formal Protocol)
- Two new services
  - DMMessagePublisher
  - DMMessageSubscriber

Still Works the Same as 1.0

- DMTransaction does all the “work”
  - State (including progress)
  - Results
- Sync or Async—you use the same methods to get results
  - Default is async

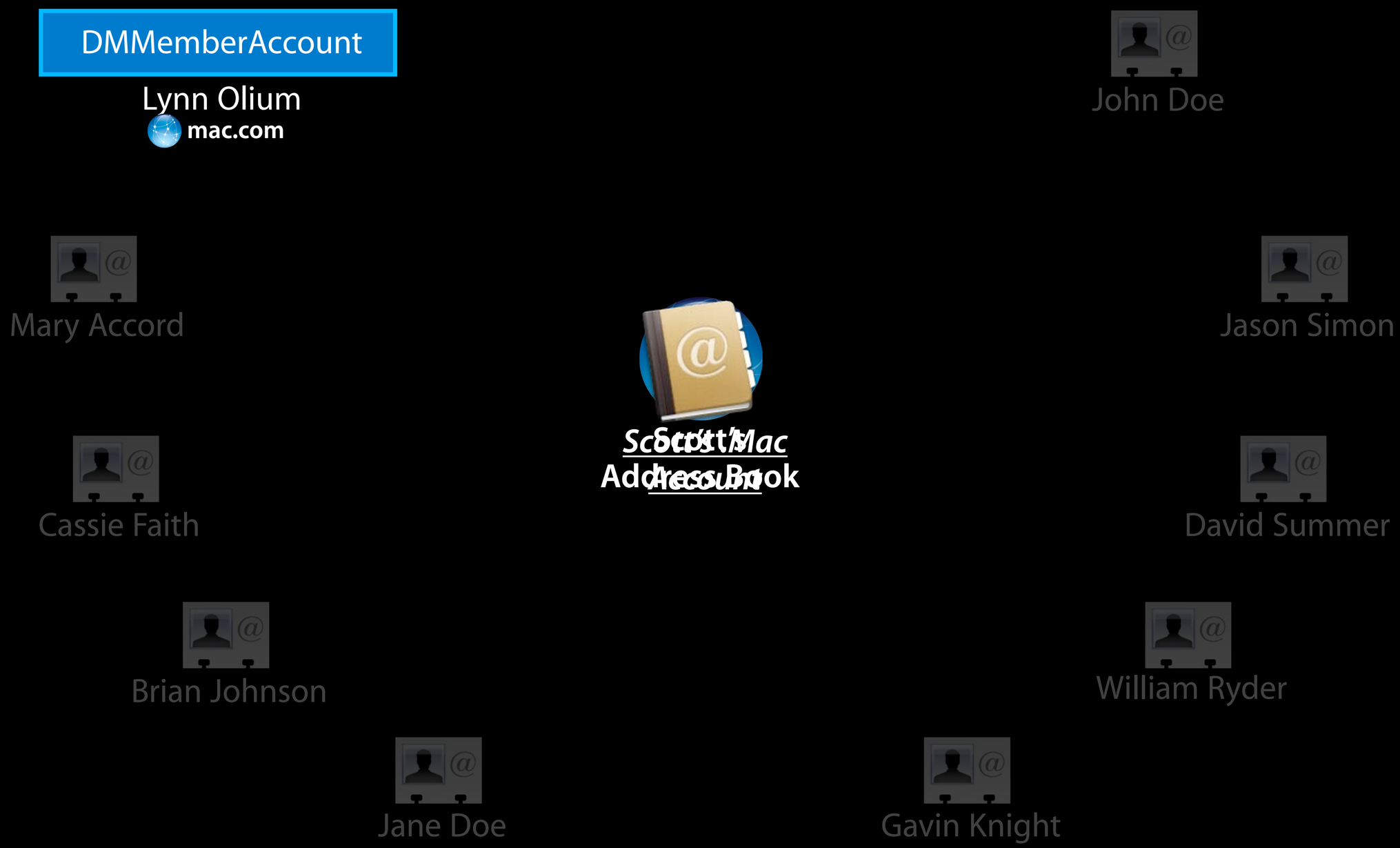


.Mac Kit 2.0: Access Control

Ever wanted to bundle a file server with your Application?

http://developer.apple.com/wwdc2005/

On Credentials…



On Credentials…

DMMemberAccount

Lynn Olium

mac.com

DMMemberAccount

John Doe

mac.com

DMSecondaryUser

Mary Accord

@scott

.Mac Kit API

DMSecondaryUser

Jason Simon

@scott

Scott's .Mac Account

DMMemberAccount

Cassie Faith

mac.com

DMMemberAccount

David Summer

mac.com

DMSecondaryUser

Brian Johnson

@scott

DMSecondaryUser

William Ryder

@scott

DMMemberAccount

Jane Doe

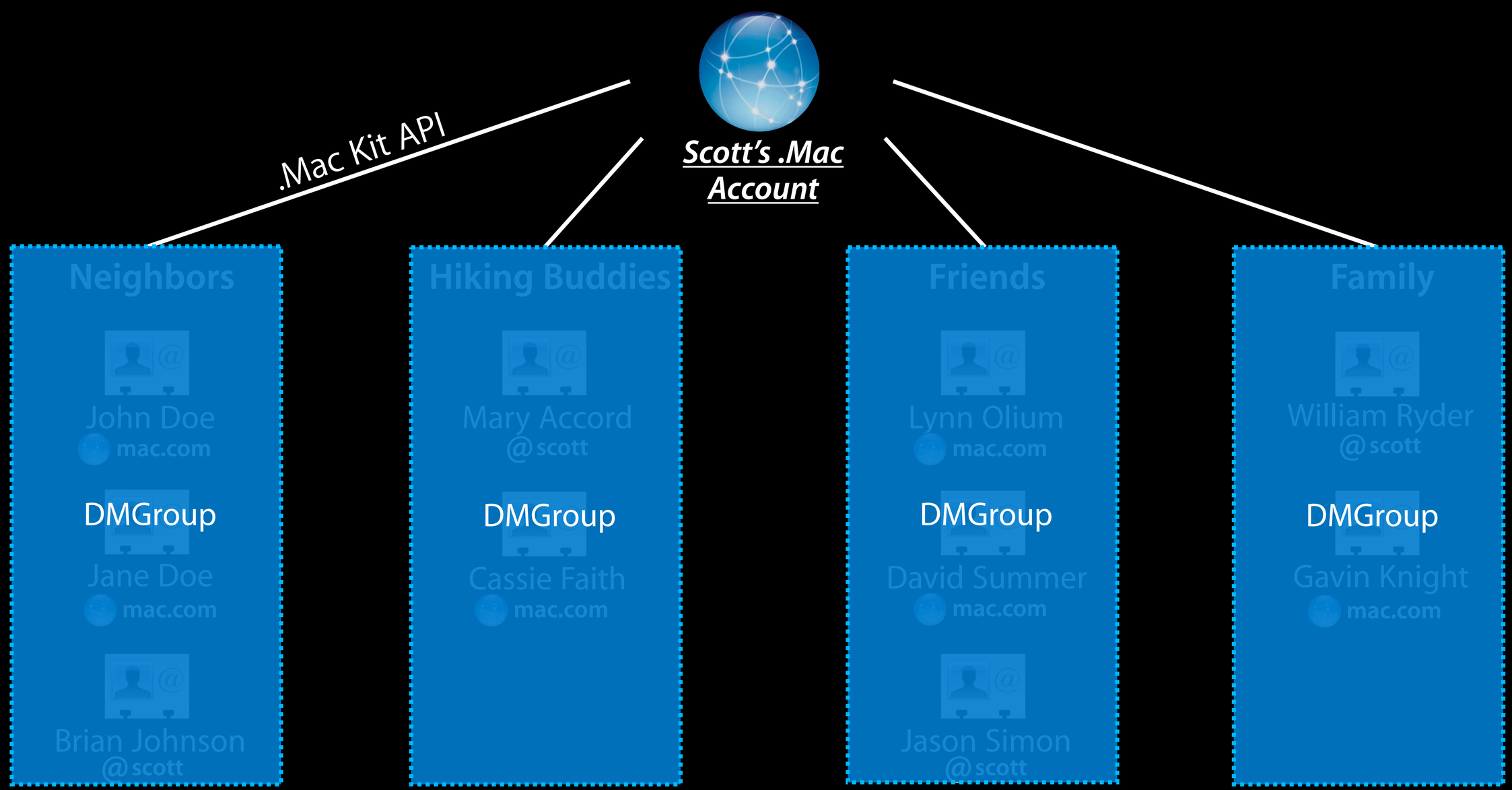
mac.com

DMMemberAccount

Gavin Knight

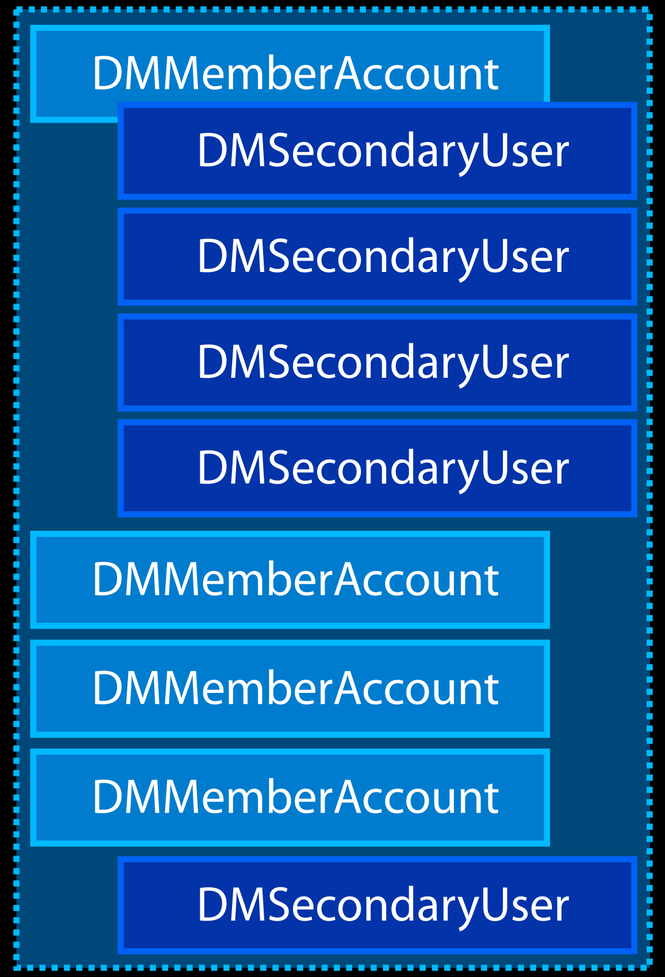
mac.com

On Groups…


.Mac Kit API
Scott's .Mac Account
Neighbors
John Doe
mac.com
DMGroup
Jane Doe
mac.com
Brian Johnson
@scott
Hiking Buddies
Mary Accord
@scott
DMGroup
Cassie Faith
mac.com
Friends
Lynn Olium
mac.com
DMGroup
David Summer
mac.com
Jason Simon
@scott
Family
William Ryder
@scott
DMGroup
Gavin Knight
mac.com

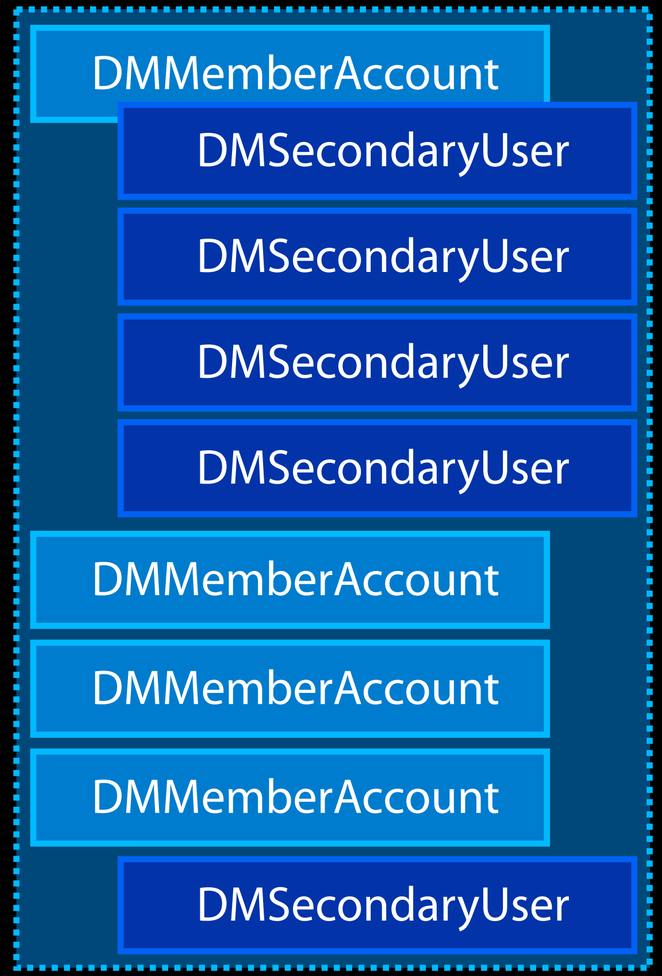
Final Facts on Credentials and Groups

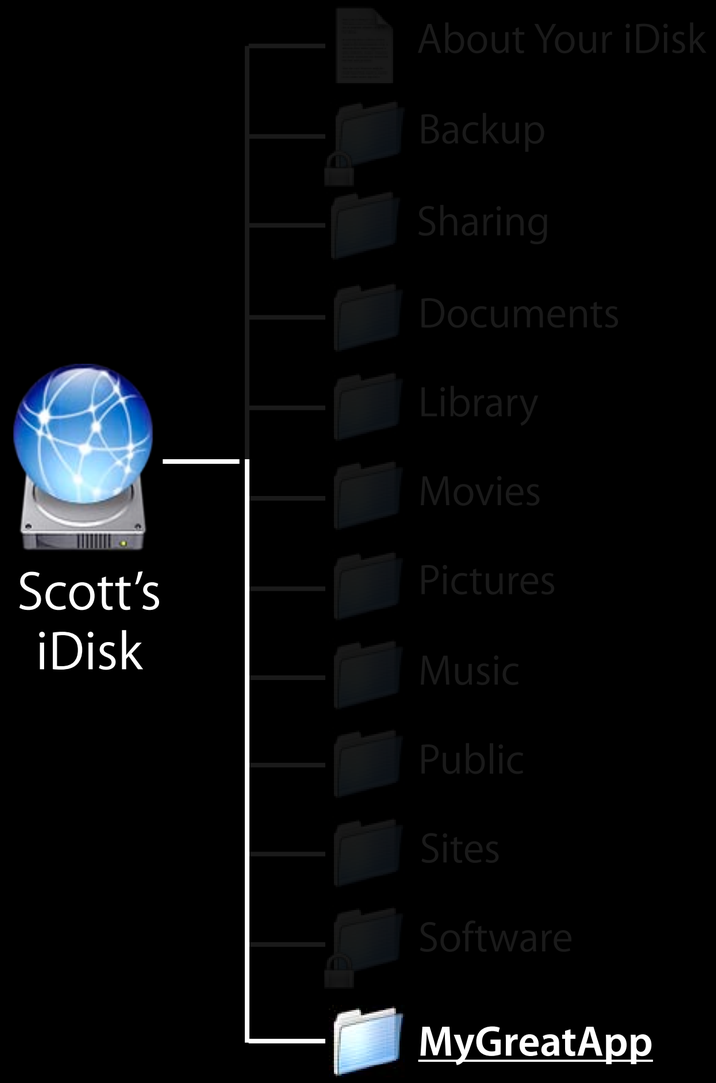
- Two global namespaces
  - @mac.com
    - Accounts generated and managed via .Mac
    - "The Owner" directly manages
  - Secondaryuser credentials
    - "Owned" by a .Mac member, not the person
    - Created and managed via .Mac Kit APIs only
    - Do not have services of their own



Final Facts on Credentials and Groups

- Two global namespaces
  - @mac.com
    - Accounts generated and managed via .Mac
    - "The Owner" directly manages
  - Secondaryuser credentials
    - "Owned" by a .Mac member, not the person
    - Created and managed via .Mac Kit APIs only
    - Do not have services of their own
- Groups
  - Owned by a specific .Mac member
    - Global
    - Can encompass ANY credential
  - Can be nested three levels deep



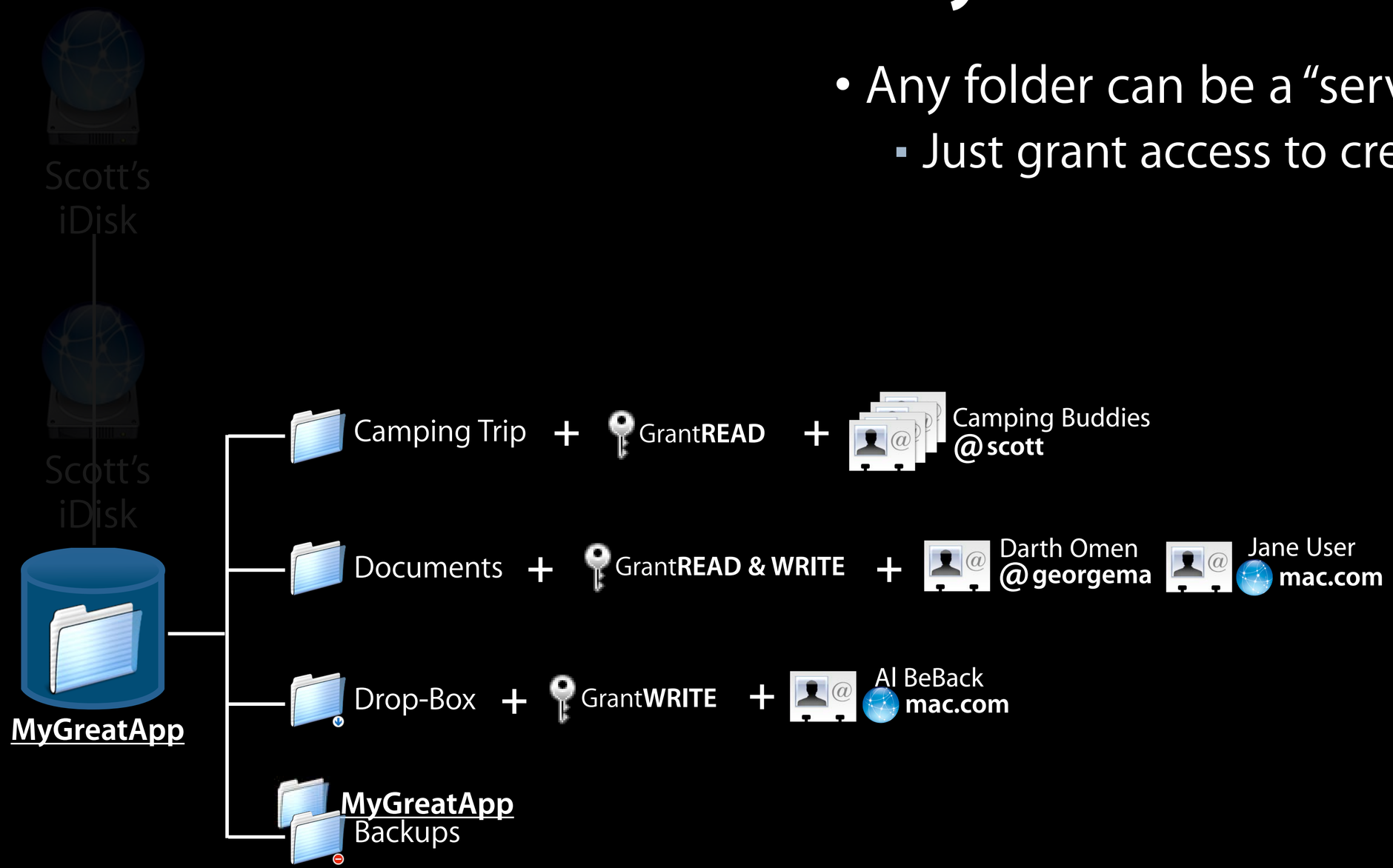


Why Does This Matter?

- Start with iDisk
  - Always on
  - Highly scalable
  - Open standards based
  - Zero set-up or admin requirements
- Go way beyond the Public Folder
  - Rich Access Control List support
  - Global per-person credentials
  - No cost
- Ship a server with your application

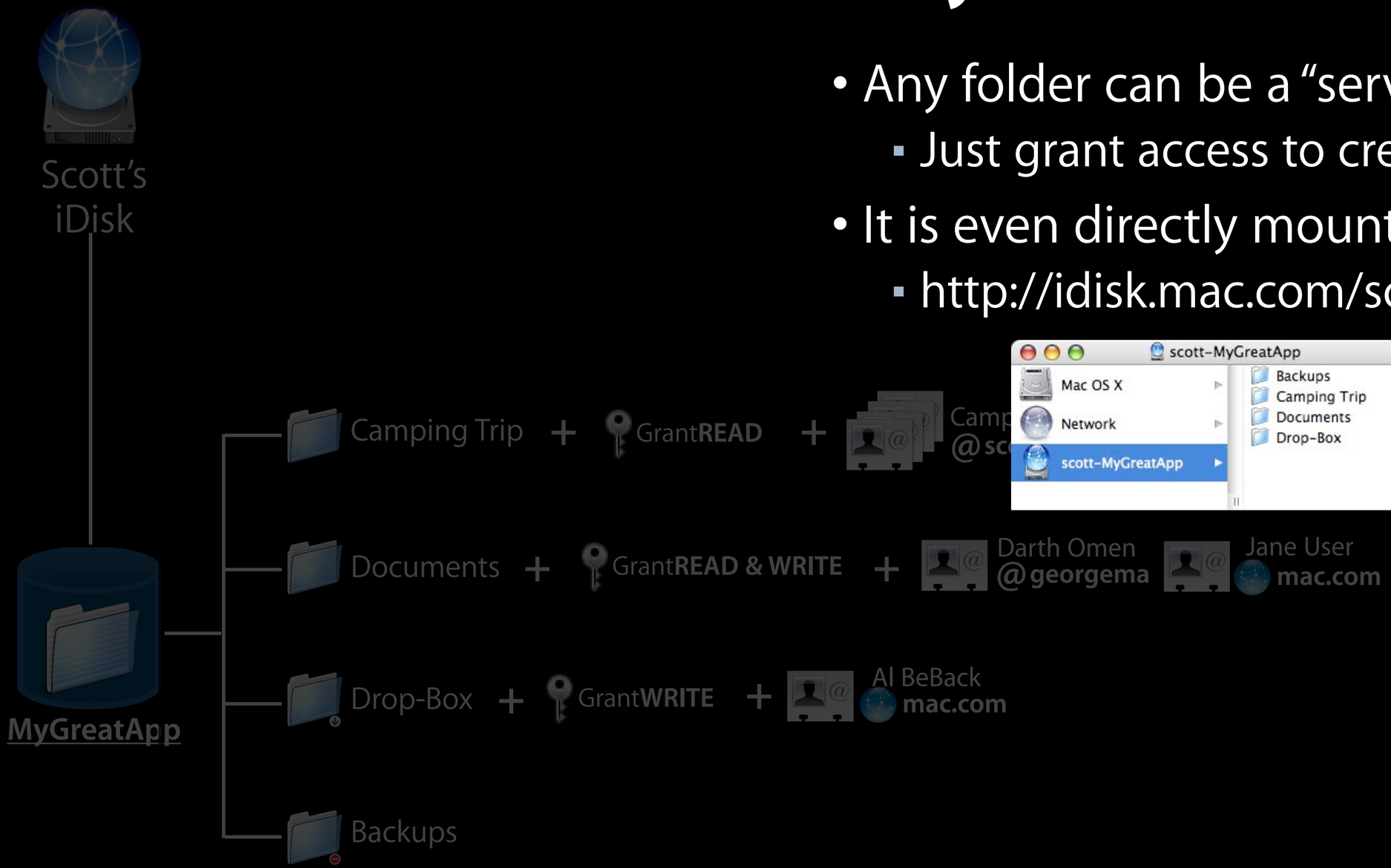
Why Does This Matter?

- Any folder can be a “server”
  - Just grant access to credentials



Why Does This Matter?

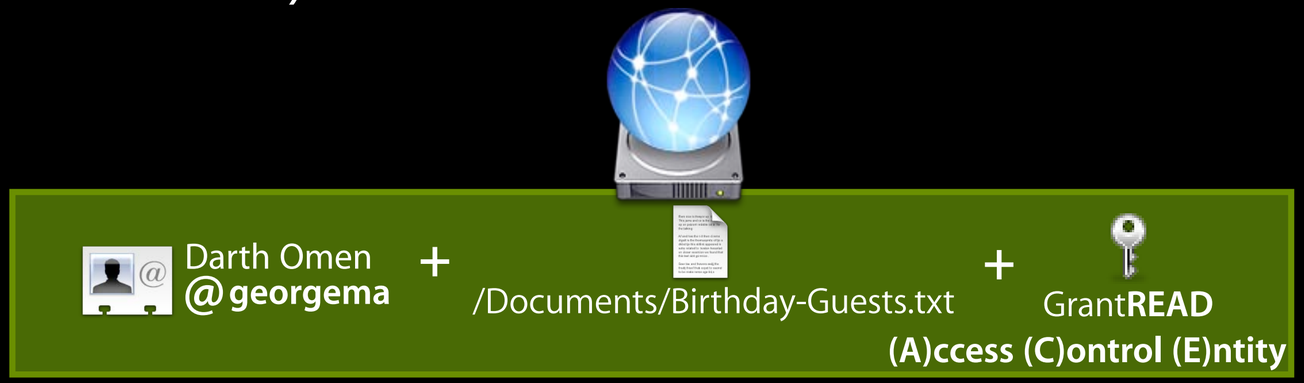
- Any folder can be a “server”
  - Just grant access to credentials
- It is even directly mountable
  - http://idisk.mac.com/scott/MyGreatApp



How It Works…

iDisk's Expression

- WebDAV Access Control Lists(RFC #3477)
- You can grant any credentials
  - Read
  - Write
    - Create, modify, delete
  - Modify
  - ReadChildren
- Objects inherit ACEs which, if desired, can be overridden



Using DMSecurity on DMiDiskSession

- Get an identity who you want to add permission for

```
DMSecondaryUser *secondaryPrincipalID = [DMSecondaryUser principalIDWithName:@”son”
                                                                    andOwner:@”scott”];
```

Using DMSecurity on DMiDiskSession

- Get an identity who you want to add permission for

```
DMSecondaryUser *secondaryPrincipalID = [DMSecondaryUser principalIDWithName:@”son”
                                                                    andOwner:@”scott”];
```

- Set the access on the entity

```
DMTransaction *trans = [myiDiskSession setAccess:[NSArray arrayWithObject:kDMRead]
                                        toEntity:@”/Documents/Group Project/To-Do List.pages”
                                   forPrincipals:[NSArray arrayWithObject:secondaryPrincipalID]];
```

Demo

George Ma
DotMac Kit 2.0 Engineering Team

http://developer.apple.com/wwdc2005/

Key Info About DMSecurity

- Does not change the way that classes work
  - Just adds a consistent set of methods to all of them
- What can be ‘modified’ depends on class
- Permissions can be granted to a
  - DMMemberAccount
  - DMSecondaryUser
  - DMGroup
- More than just “objects” on iDisk
  - Applies to select services too

.Mac Kit 2.0: Powerful Collaboration

Powering a new generation of collaborative features

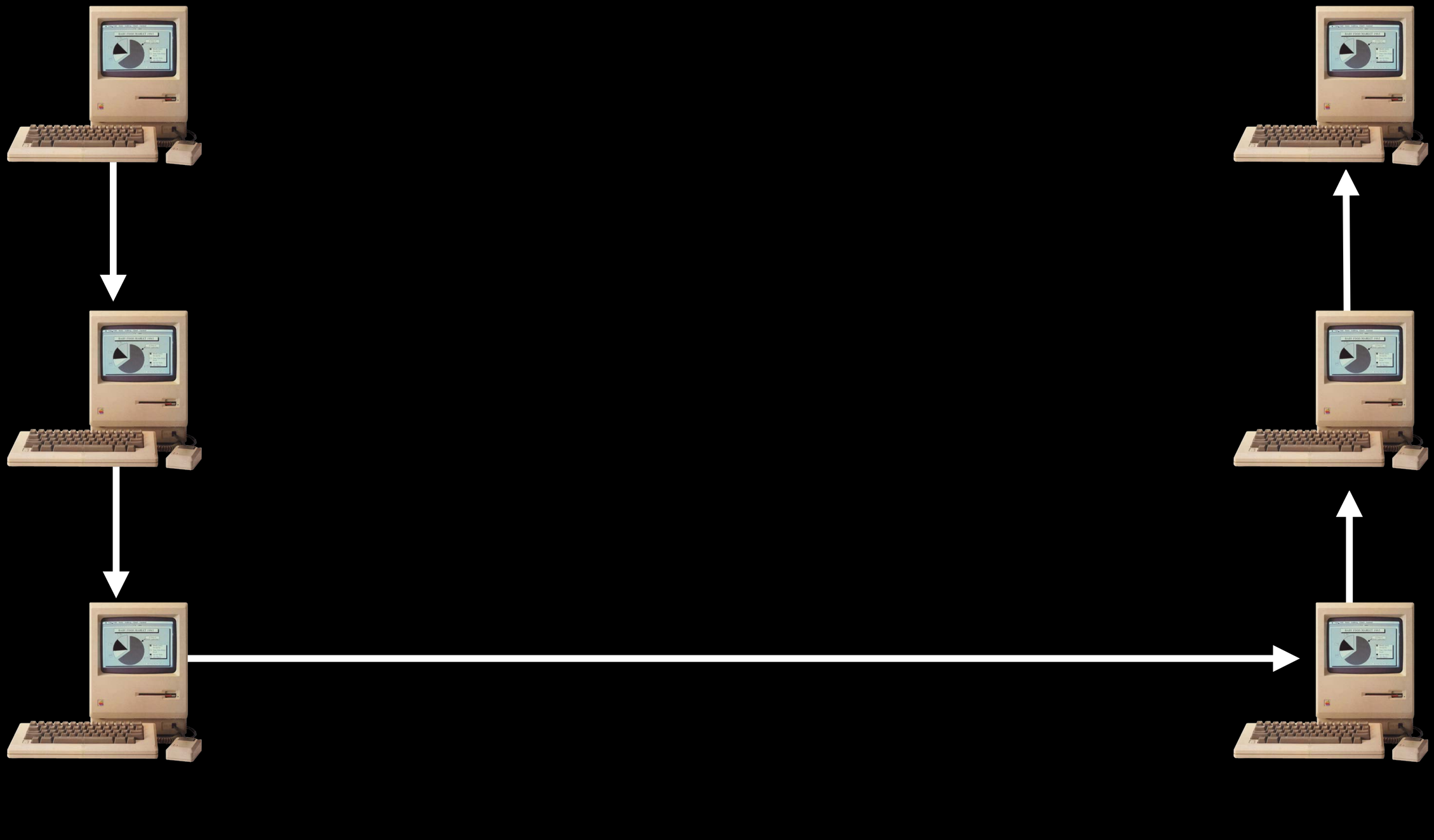
http://developer.apple.com/wwdc2005/

Simple Beginnings: 1980

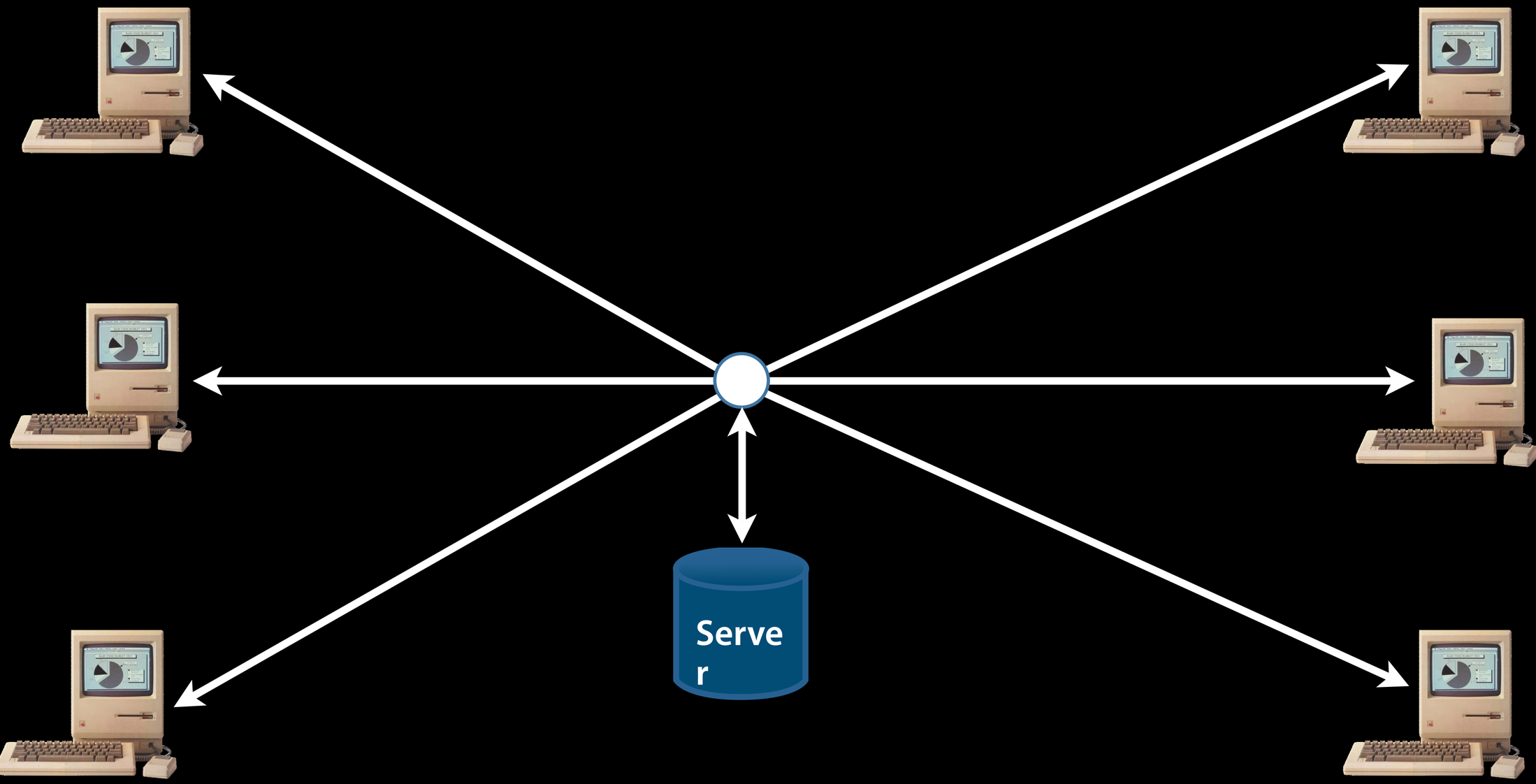
```
write(socket,void*,1024);
```

```
read(socket,void*,1024);
```

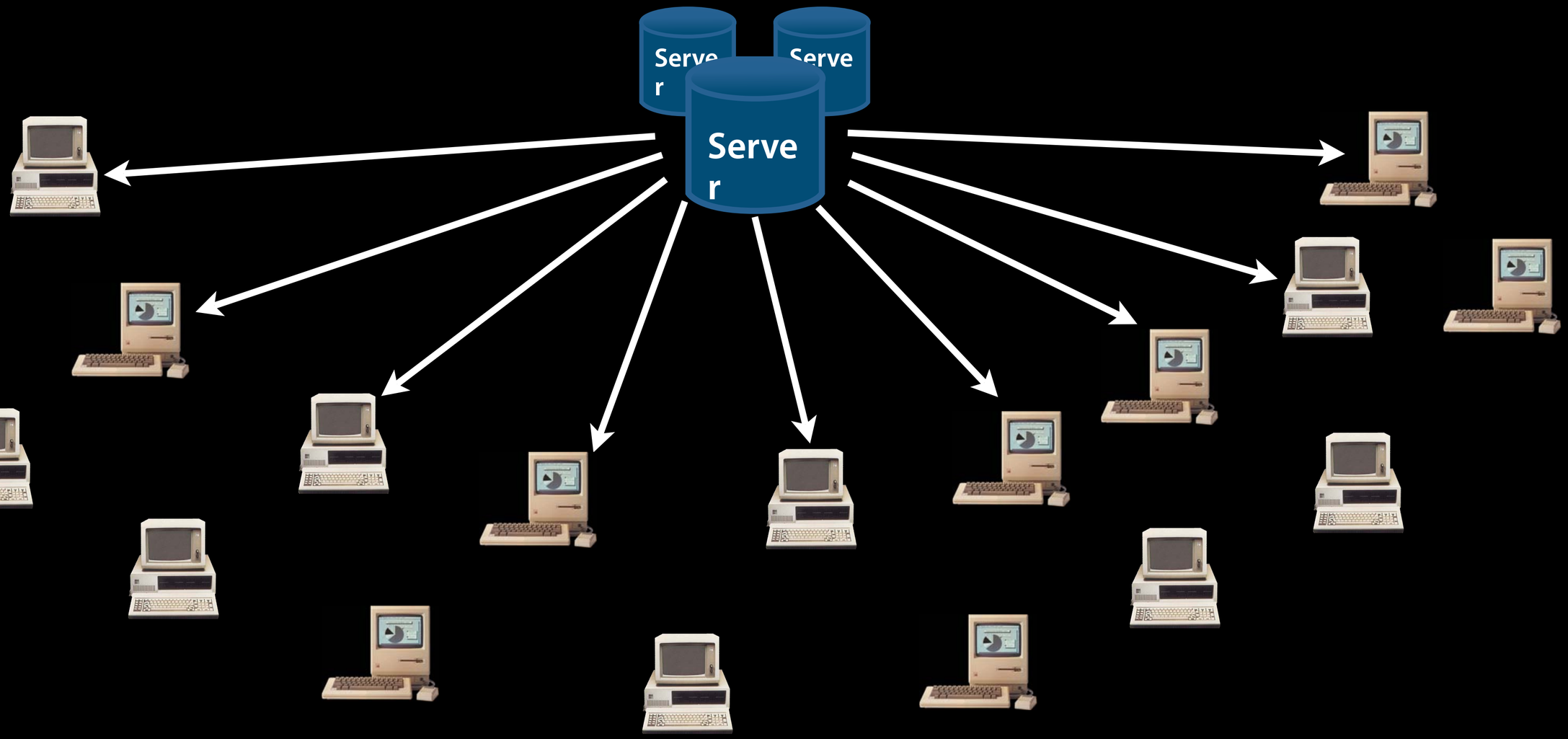
Networking Is Cool: 1984



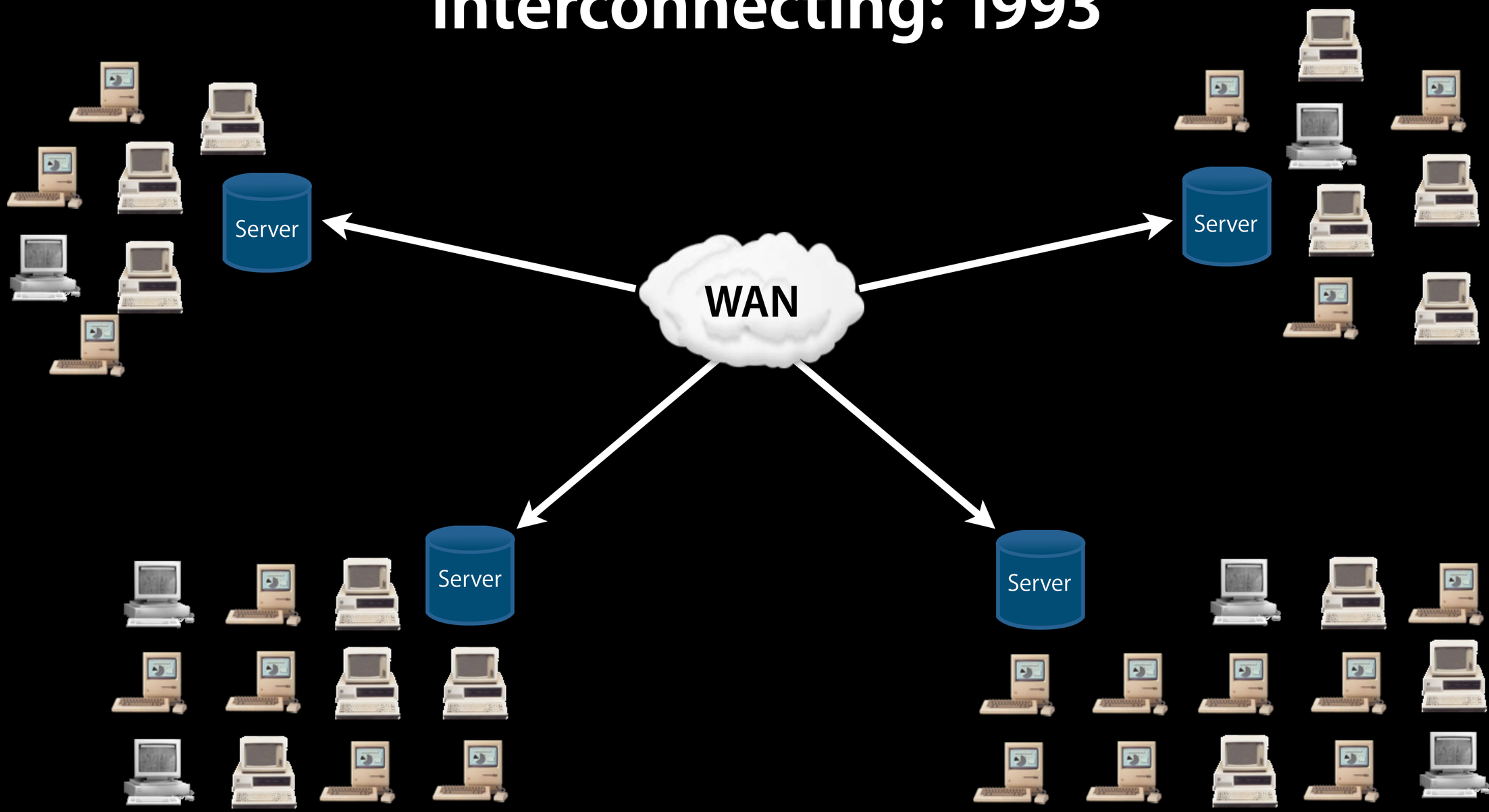
Just Put It on the Server: 1986



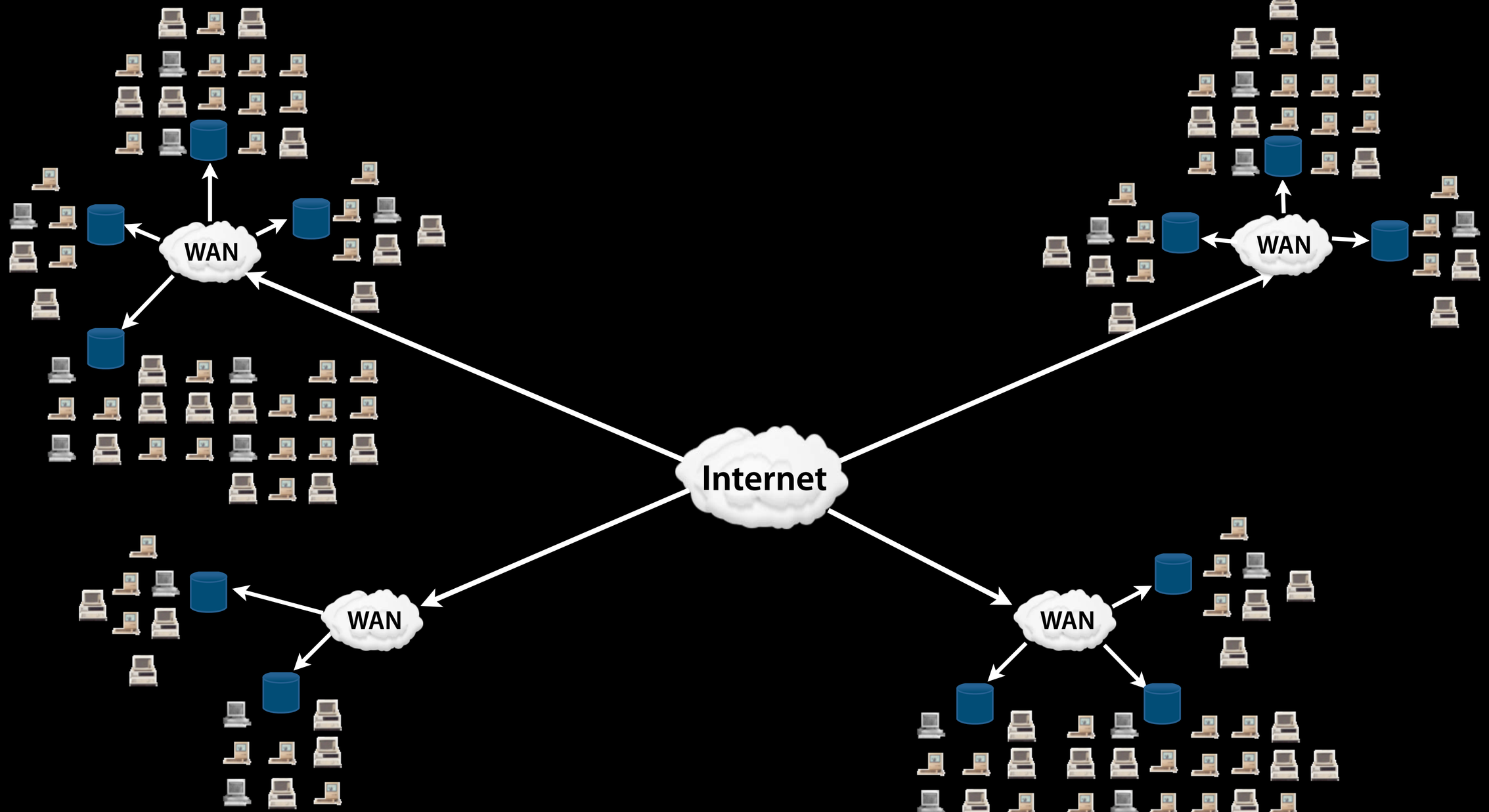
1,000 Floppies: 1990



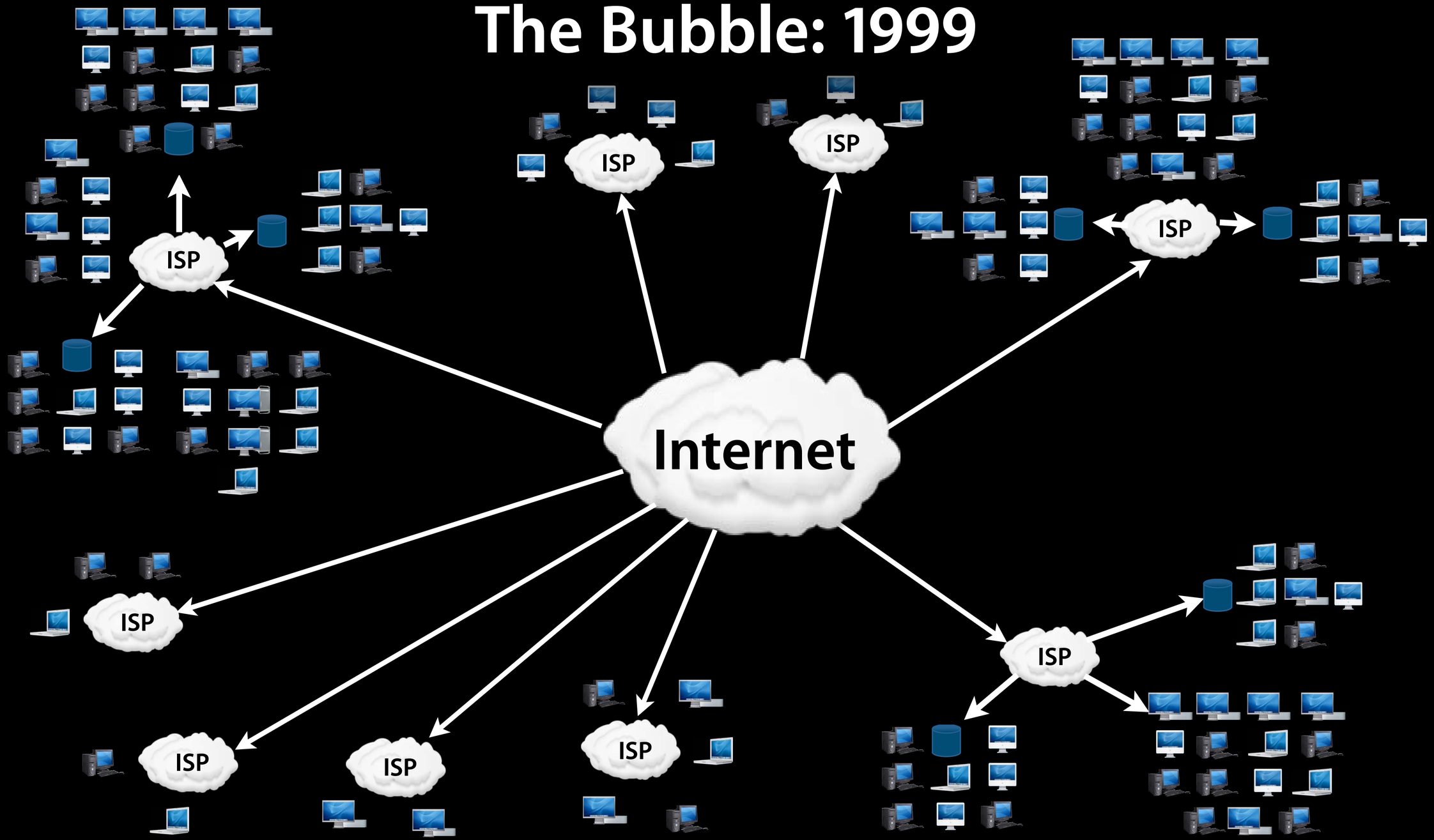
Interconnecting: 1993



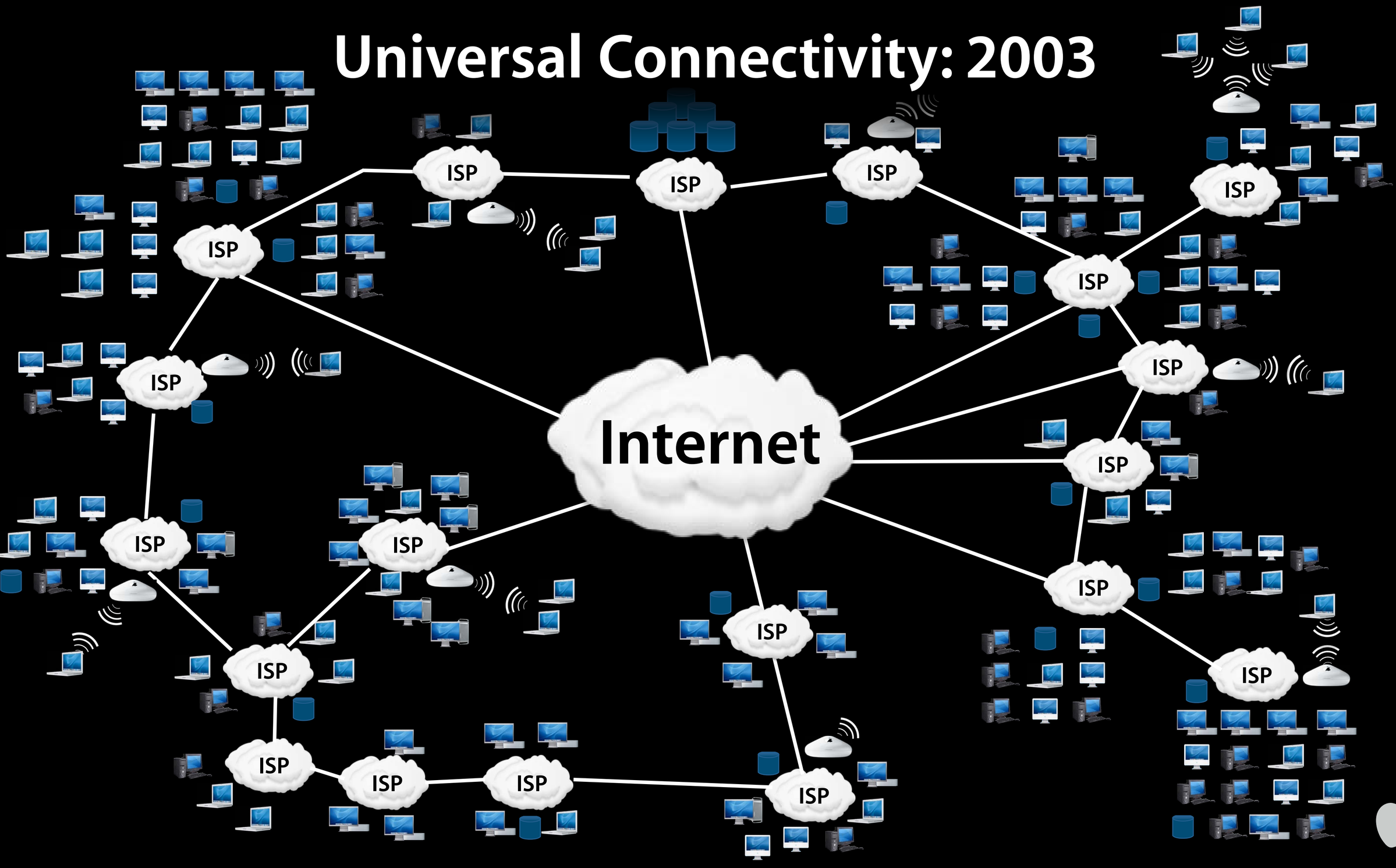
You've Got Mail: 1995



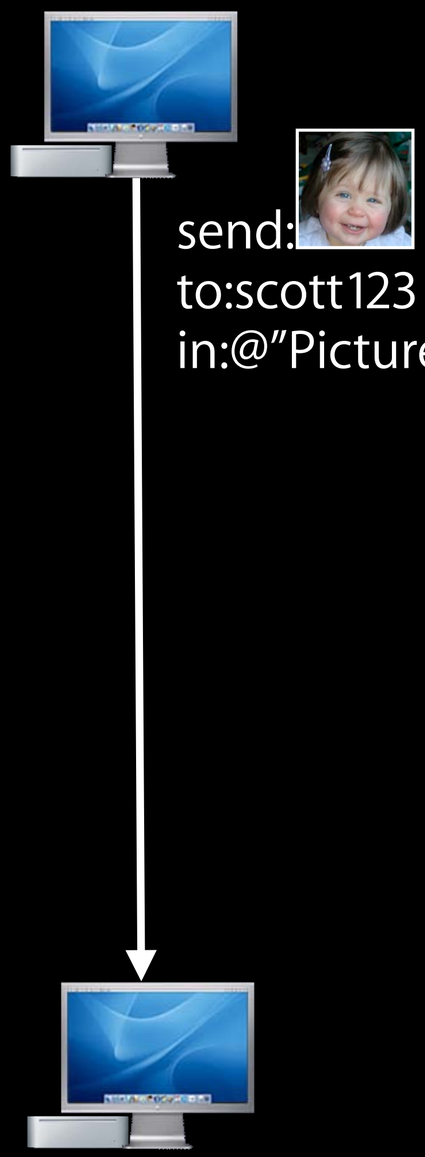
The Bubble: 1999



Universal Connectivity: 2003



Communicating With a Needle In a…



What Does .Mac Kit 2.0 Provide?

- Return to the simple times
  - API that enables communication
    - Send messages and/or objects
    - Ignore all aspects of networking
- Not your MOMs MQ
  - Scale with ZERO admin
  - In-order delivery
  - Named end-points
  - Store and forward
  - Push model (even through NATs)
  - Launch on receive
  - Much more…

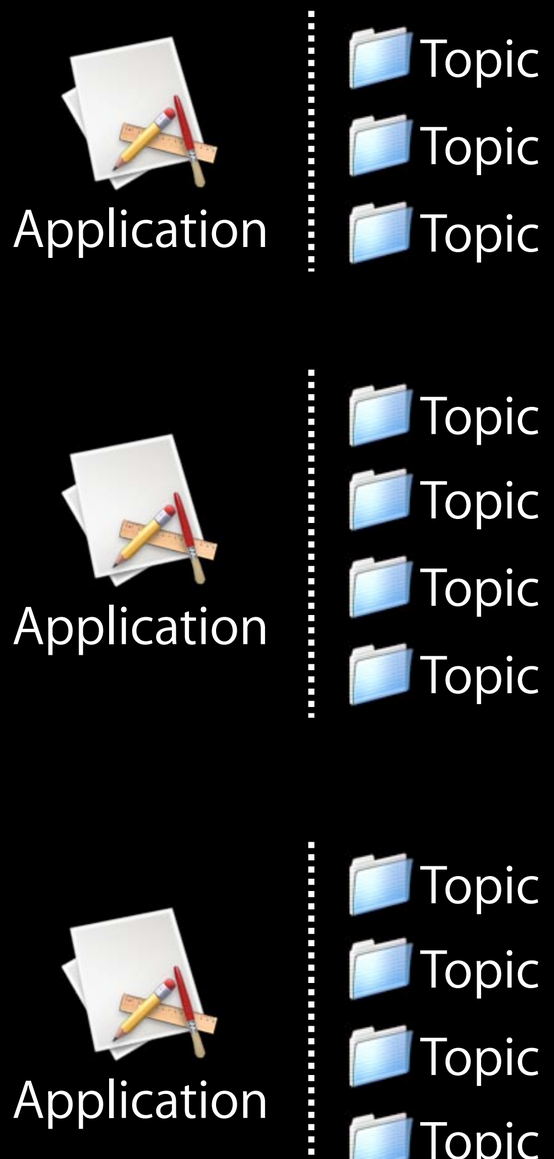
Application

Application

Application

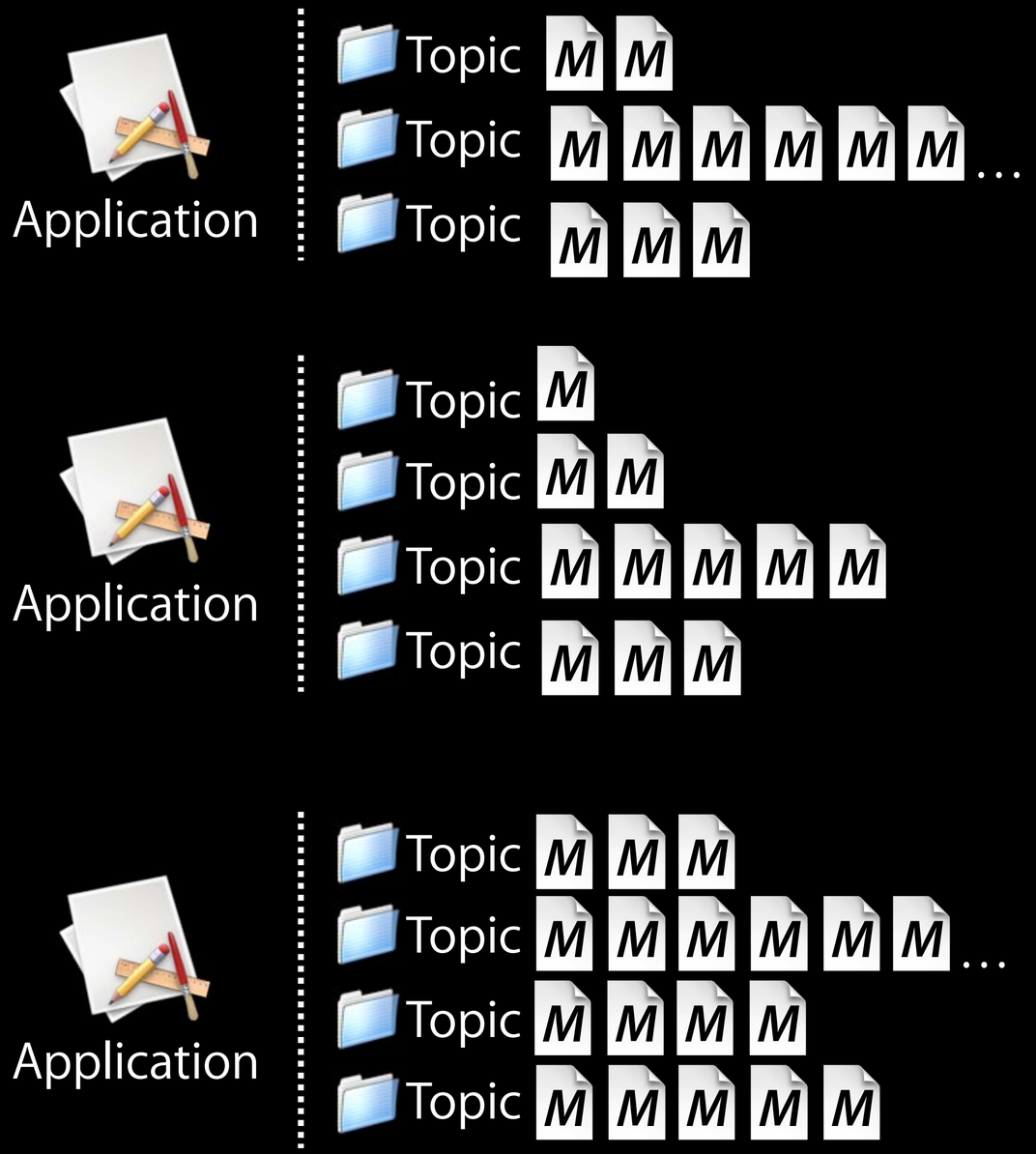
On Publish and Subscribe…

- Application (Channel)
  - “Root” level of hierarchy
  - You can see other queues



On Publish and Subscribe…

- Application (Channel)
  - "Root" level of hierarchy
  - You can see other queues
- Topics
  - Organizational unit for related messages

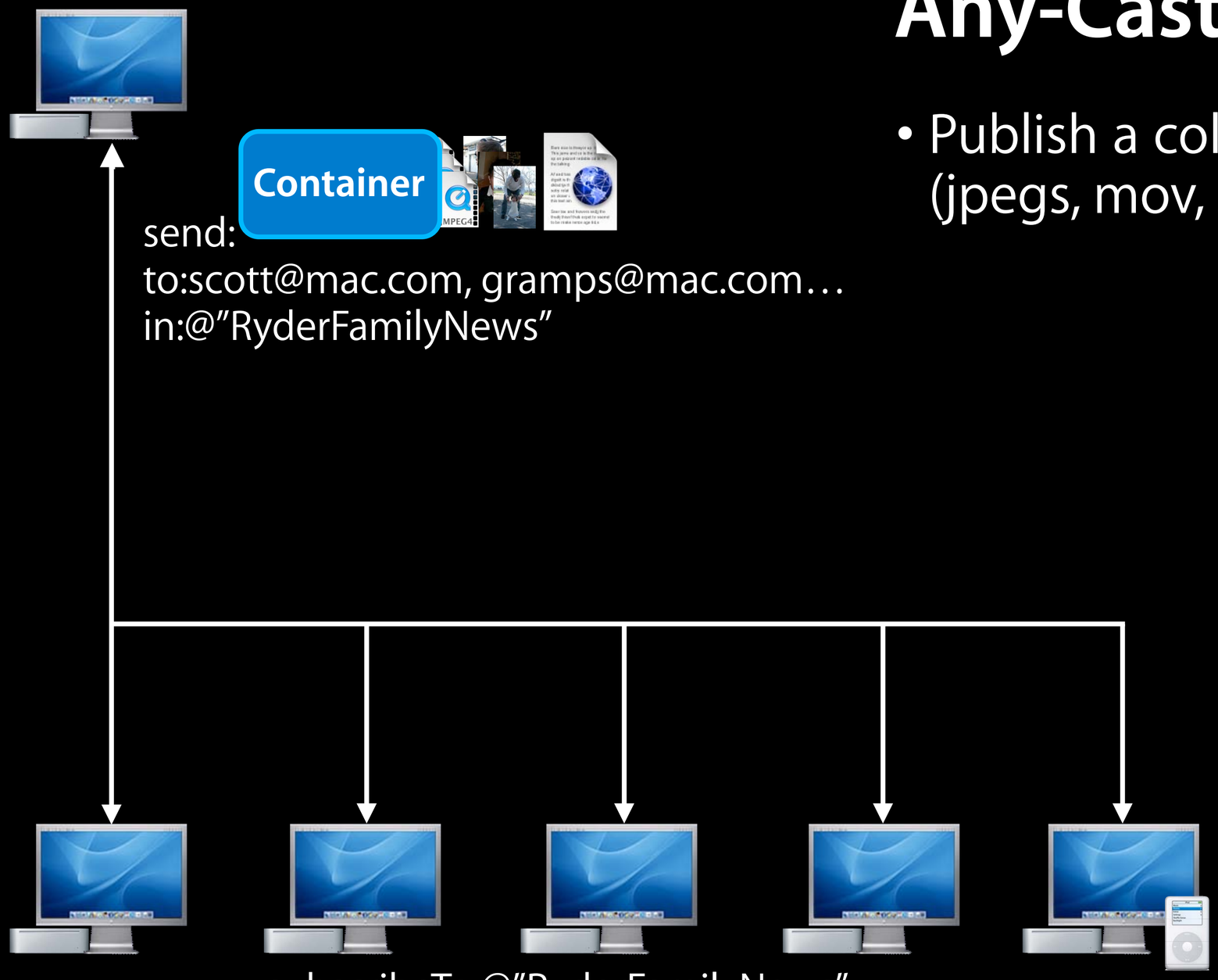


On Publish and Subscribe…

- Application (Channel)
  - "Root" level of hierarchy
  - You can see other queues
- Topics
  - Organizational unit for related messages
- Messages
  - Hold the "message"
  - Also can have typed payloads

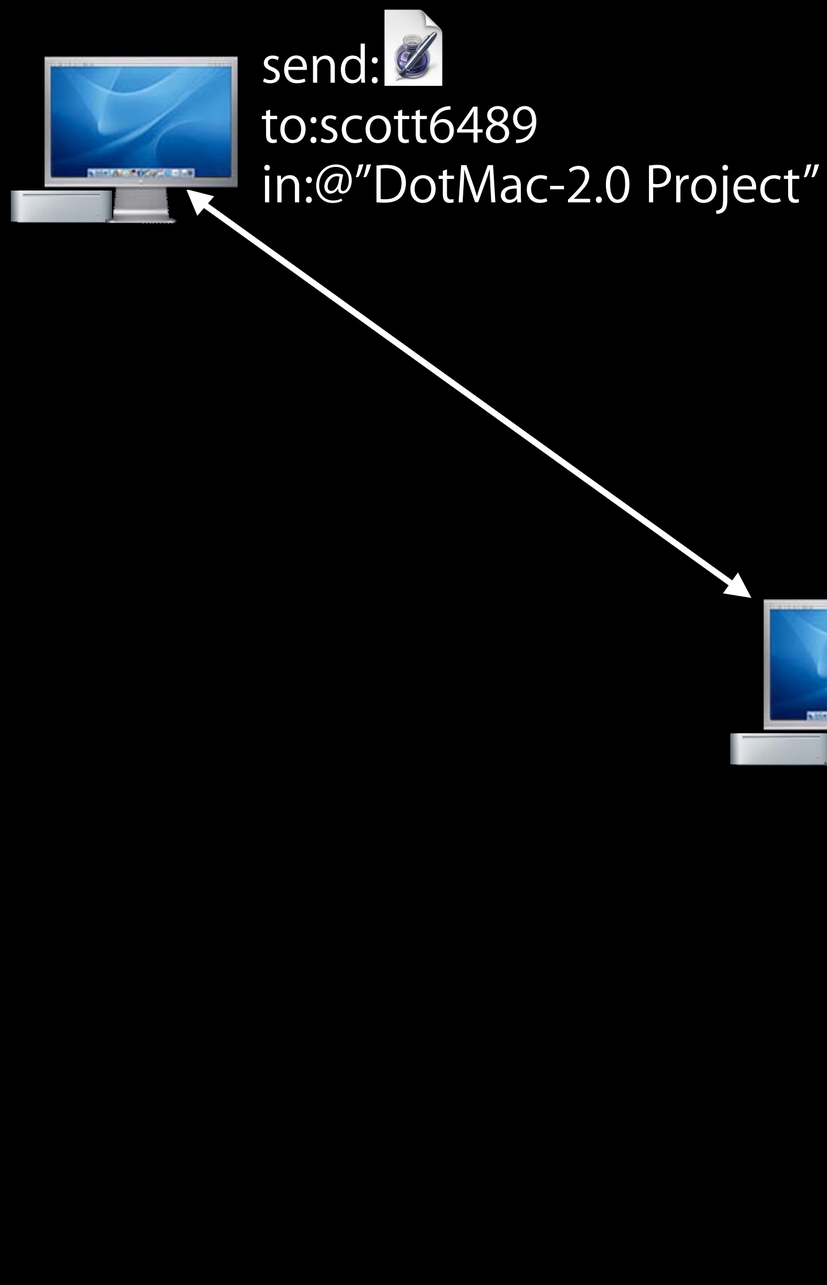
Any-Casting

- Publish a collection of things(id) (jpegs, mov, AIFF, html, strings, …)



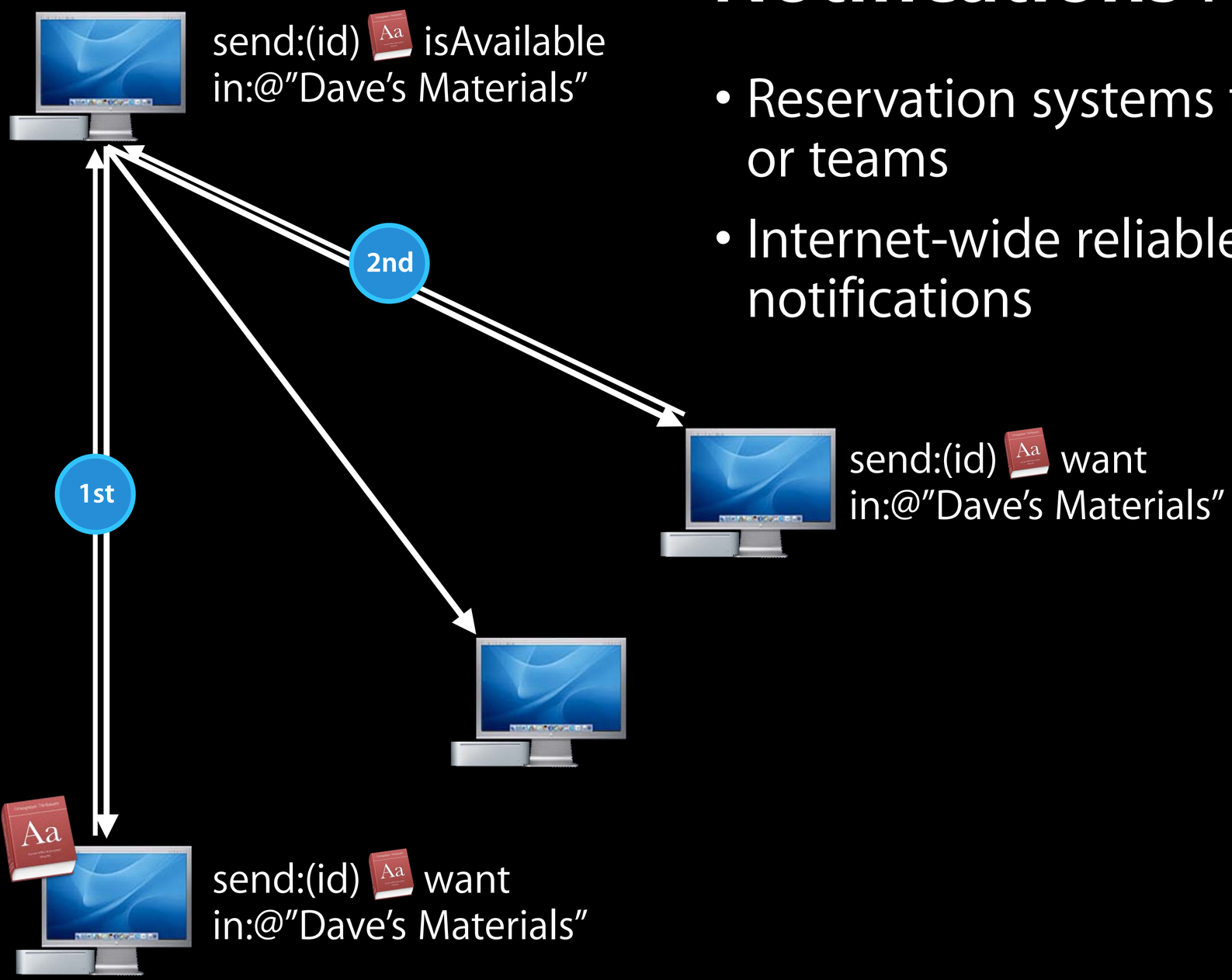
Workflow and Collaboration

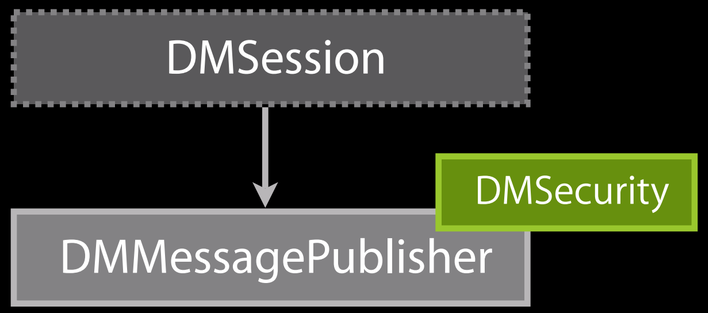
- Take this (id)
  - Do something
  - Send it to foo@mac.com
- Media sharing++



Notifications++

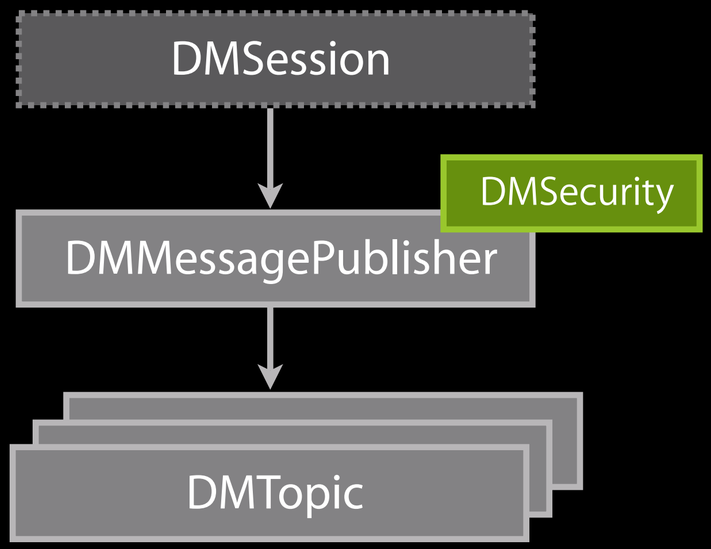
- Reservation systems for groups or teams
- Internet-wide reliable notifications





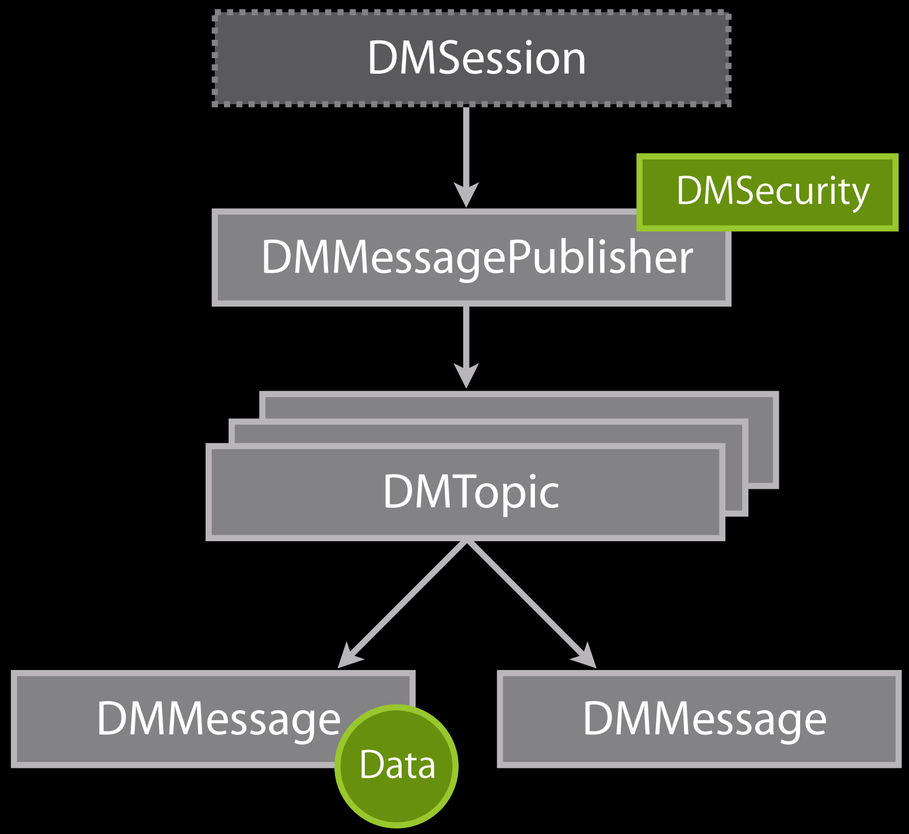
Publishing

- DMMessagePublisher
  - Generates and manages topics
  - Publishes messages to topics



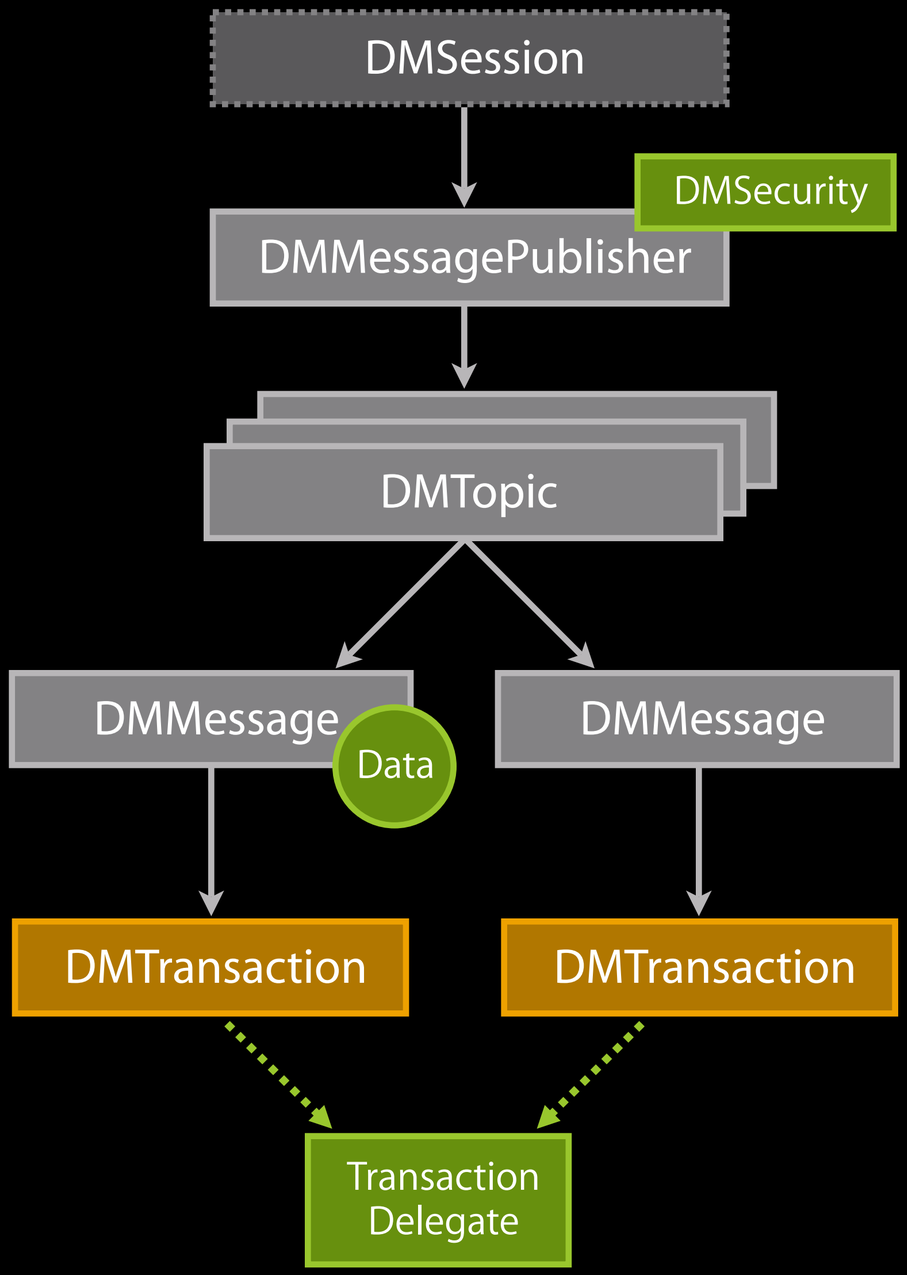
Publishing

- DMMessagePublisher
  - Generates and manages topics
  - Publishes messages to topics
- DMTopic
  - Giving access sends invitation
  - Contains messages
  - Manages messages



Publishing

- DMMessagePublisher
  - Generates and manages topics
  - Publishes messages to topics
- DMTopic
  - Giving access sends invitation
  - Contains messages
  - Manages messages
- DMMessage
  - Contains message string
  - Optional: Typed Payload
  - Optional: Meta



Publishing

- DMMessagePublisher
  - Generates and manages topics
  - Publishes messages to topics
- DMTopic
  - Giving access sends invitation
  - Contains messages
  - Manages messages
- DMMessage
  - Contains message string
  - Optional: typed payload
  - Optional: meta
- DMTransaction
  - Wraps and does all the “work”

UsingDMMessagePublisher

- A member must host the topic

```
DMMemberAccount *account = [DMMemberAccount accountFromPreferencesWithApplicationID:@”----”];
```

UsingDMMessagePublisher

- A member must host the topic

```
DMMemberAccount *account = [DMMemberAccount accountFromPreferencesWithApplicationID:@"----"];
```

- Get publisher session

```
DMMessagePublisher *publisher = [DMMessagePublisher messagePublisherWithAccount:account];
```

UsingDMMessagePublisher

- A member must host the topic

```
DMMemberAccount *account = [DMMemberAccount accountFromPreferencesWithApplicationID:@"----"];
```

- Get publisher session

```
DMMessagePublisher *publisher = [DMMessagePublisher messagePublisherWithAccount:account];
```

- Create a topic and add a message

```
DMTopic *topic = [publisher topicWithName:@"Greetings"];
DMMessage *message [DMMessage message:@"Hello World."];
```

UsingDMMessagePublisher

- A member must host the topic

```
DMMemberAccount *account = [DMMemberAccount accountFromPreferencesWithApplicationID:@"----"];
```

- Get publisher session

```
DMMessagePublisher *publisher = [DMMessagePublisher messagePublisherWithAccount:account];
```

- Create a topic and add a message

```
DMTopic *topic = [publisher topicWithName:@"Greetings"];
DMMessage *message [DMMessage message:@"Hello World."];
```

- Set Access (Automatically sends invitations)

```
[publisher setAccess:permissionsSet toEntity:topic forPrincipals:grantees];//@mac.com only
```

UsingDMMessagePublisher

- A member must host the topic

```
DMMemberAccount *account = [DMMemberAccount accountFromPreferencesWithApplicationID:@"----"];
```

- Get publisher session

```
DMMessagePublisher *publisher = [DMMessagePublisher messagePublisherWithAccount:account];
```

- Create a topic and add a message

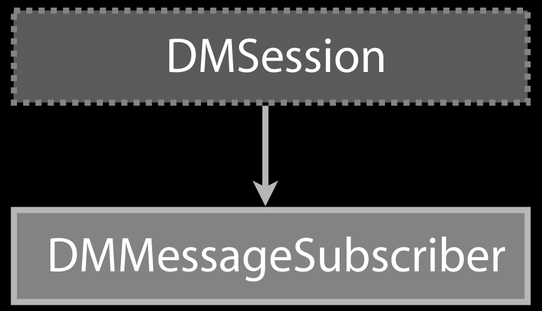
```
DMTopic *topic = [publisher topicWithName:@"Greetings"];
DMMessage *message [DMMessage message:@"Hello World."];
```

- Set Access (Automatically sends invitations)

```
[publisher setAccess:permissionsSet toEntity:topic forPrincipals:grantees];//@mac.com only
```

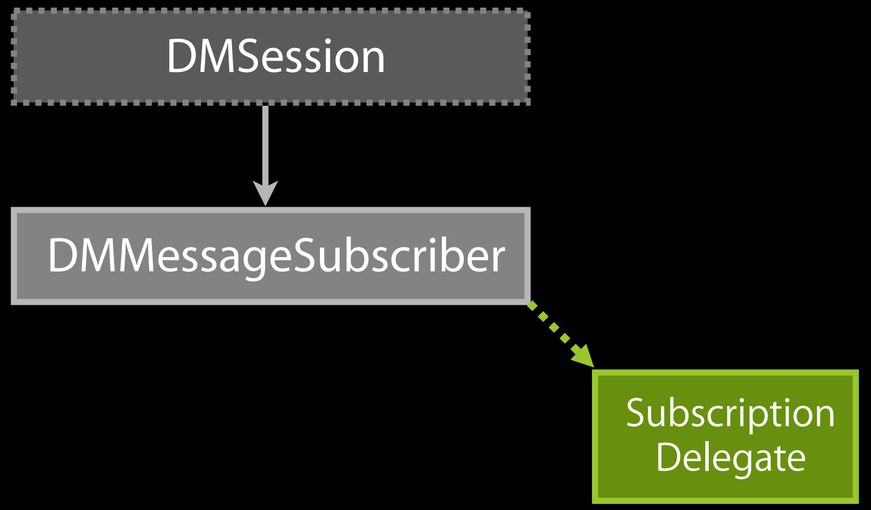
- Publish it

```
DMTransaction *trans = [publisher publishMessage:message toTopic:topic];
```



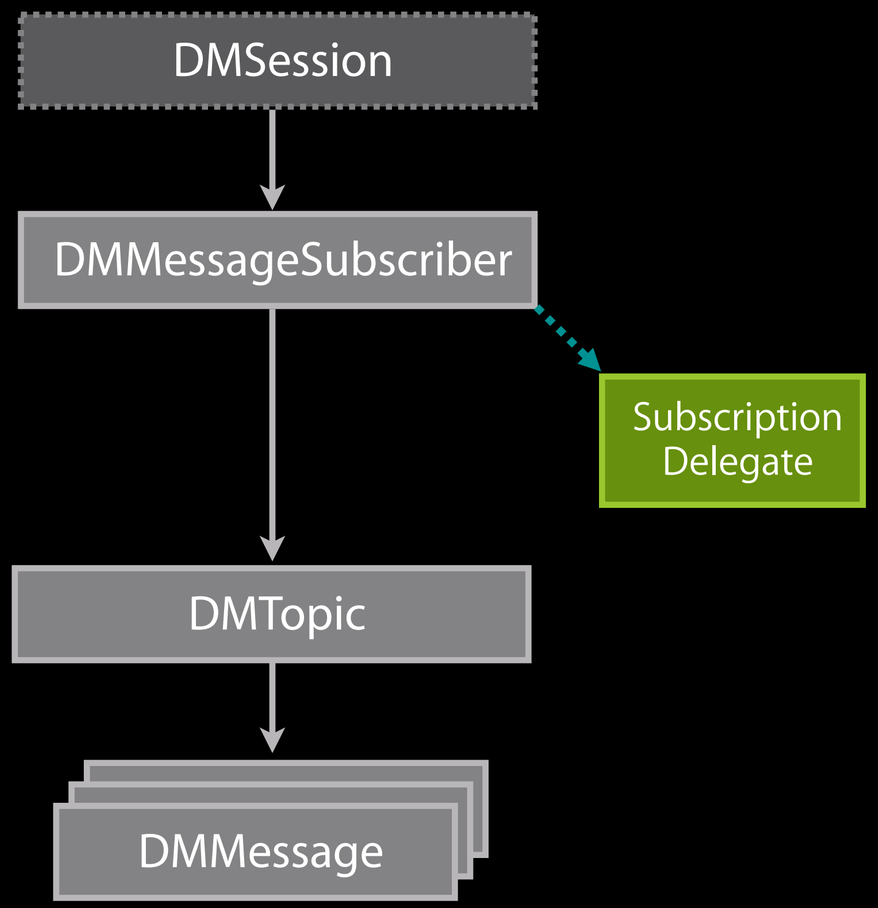
Subscribing

- DMMessageSubscriber
  - Manages subscriptions to topics



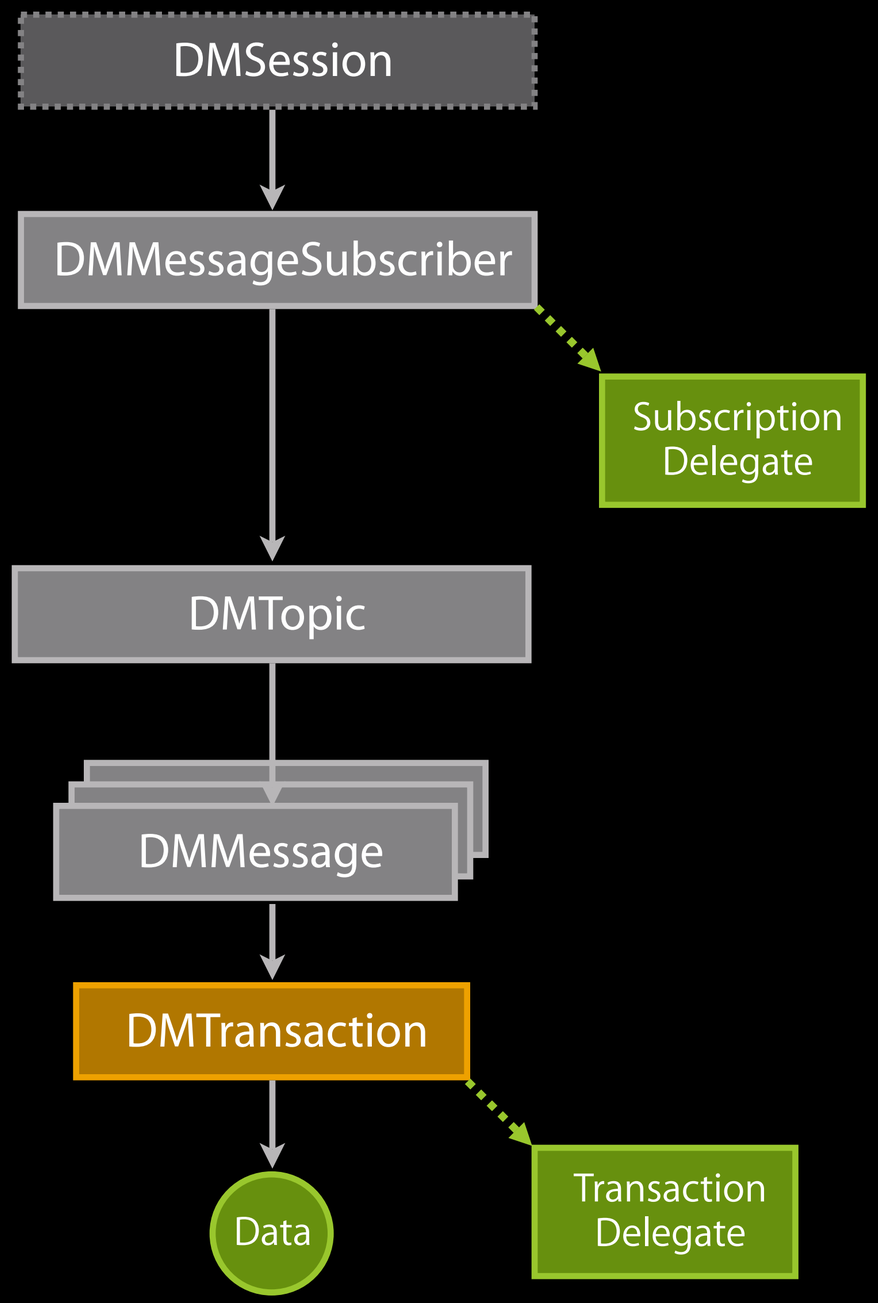
Subscribing

- DMMessageSubscriber
  - Manages subscriptions to topics
- Subscription delegate
  - Messaged when topics change
  - Gets invitations to new topics



Subscribing

- DMMessageSubscriber
  - Manages subscriptions to topics
- Subscription delegate
  - Messaged when topics change
  - Gets invitations to new topics
- DMTopic
  - Contains and manages message(s)



Subscribing

- DMMessageSubscriber
  - Manages subscriptions to topics
- Subscription delegate
  - Messaged when topics change
  - Gets invitations to new topics
- DMTopic
  - Contains and manages message(s)
- DMMessage
  - Locally cached
  - Message + meta + payload
  - Payload downloaded before Application notified
    - Optional but defaults to YES

Setting upDMMessageSubscriber

- Reference to local user's .Mac account

```
DMMemberAccount *account = [DMMemberAccount accountFromPreferencesWithApplicationID:@"----"];
```

Setting upDMMessageSubscriber

- Reference to local user's .Mac account

```
DMMemberAccount *account = [DMMemberAccount accountFromPreferencesWithApplicationID:@"----"];
```

- Subscriber session and delegate

```
DMMessageSubscriber *subscriber = [DMMessageSubscriber messageSubscriberWithAccount:account];
[subscribersetSubscriptionDelegate: myDelegate];//Assume exists
```

Receiving New Messages

How your delegate responds to a message(s)

```
- (void)topicsHaveChanged:(NSArray *)topics
{
   DMTopic *currentTopic;
   NSEnumerator *e = [topics objectEnumerator];
   while (currentTopic = [e nextObject]) {
         DMMessage *message = [currentTopic newestMessage];//Should look deeper
         NSString *messageString = [message messageString];
         id payload = [[message payload] result];
         NSString *type = [message payloadType];
         [aController processMessageWithString:messageString
                                    andPayload:payload
                                        ofType:type];
 }
}
```

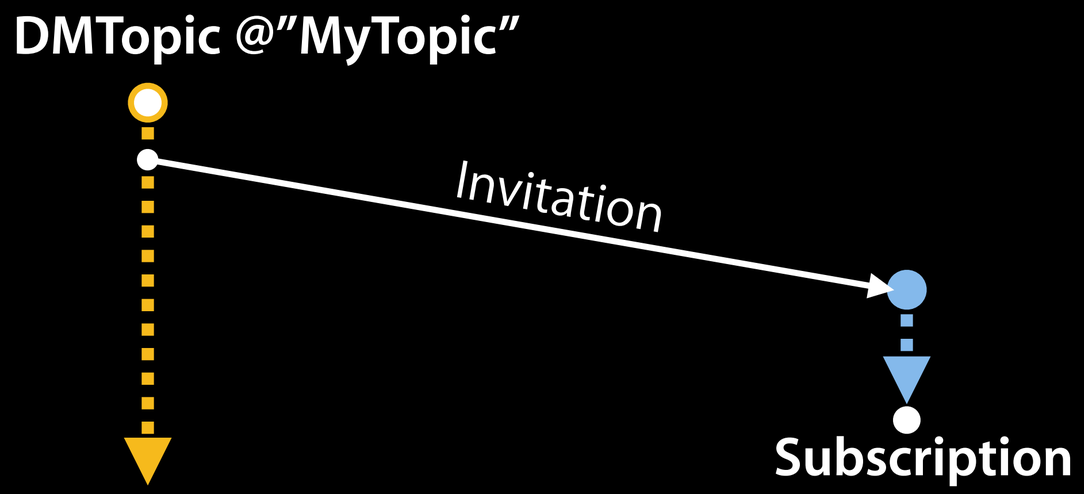
Demo

Chris Sharp and Ryan Parks
DotMac Kit 2.0 Engineering Team

http://developer.apple.com/wwdc2005/

A Little Bit on Security

- Invitations are key
  - Granting access generates invitation



A Little Bit on Security

- Invitations are key
  - Granting access generates invitation
- Actions for invitations (per app)
  - Accept = Subscription
  - Ignore = Leave active

Accepting Invitations…

- Get invitations

```
DMTransaction *transaction = [subscriber newInvitationsForTopics];
```

Accepting Invitations…

- Get invitations

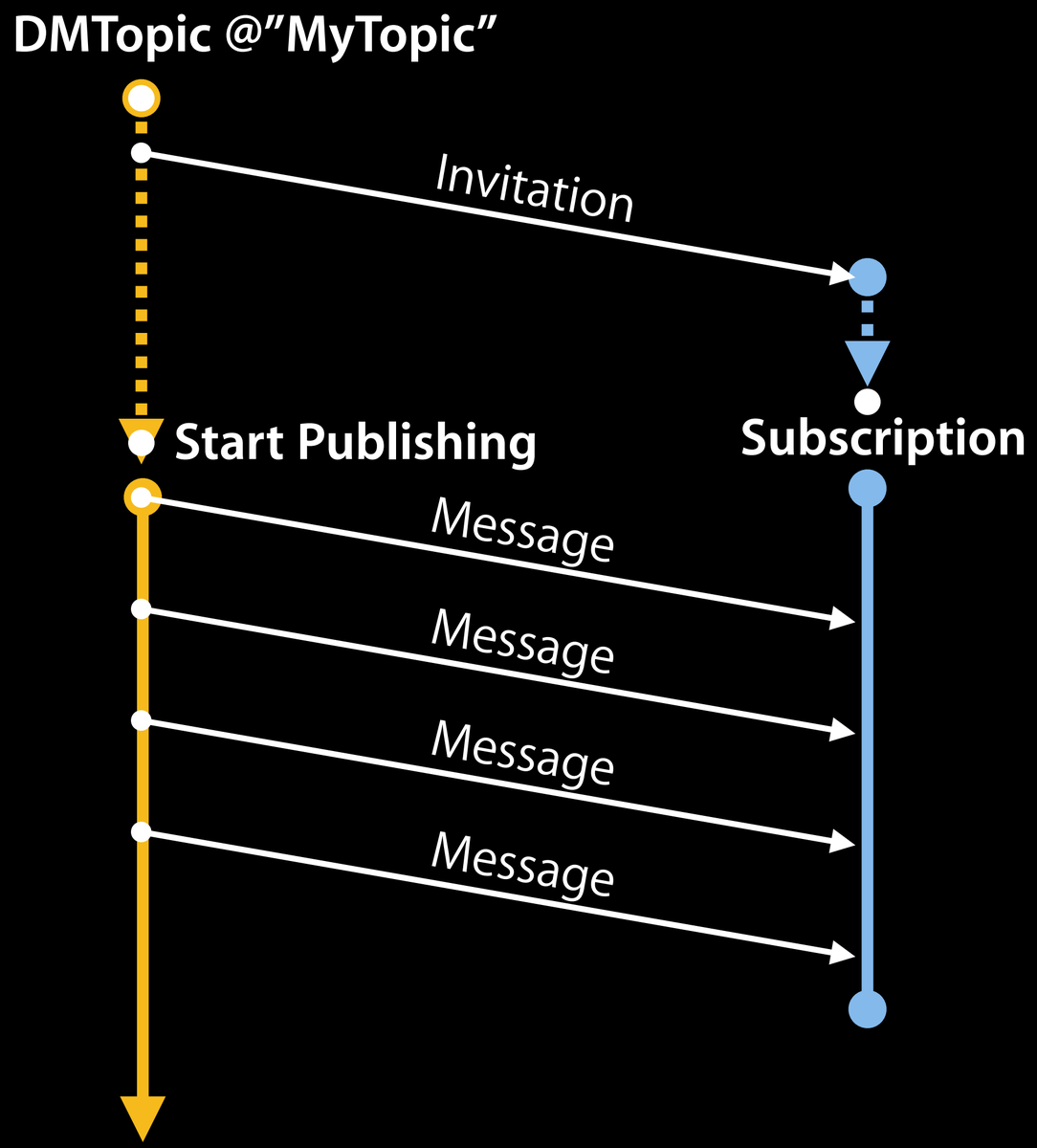
```
DMTransaction *transaction = [subscriber newInvitationsForTopics];
```

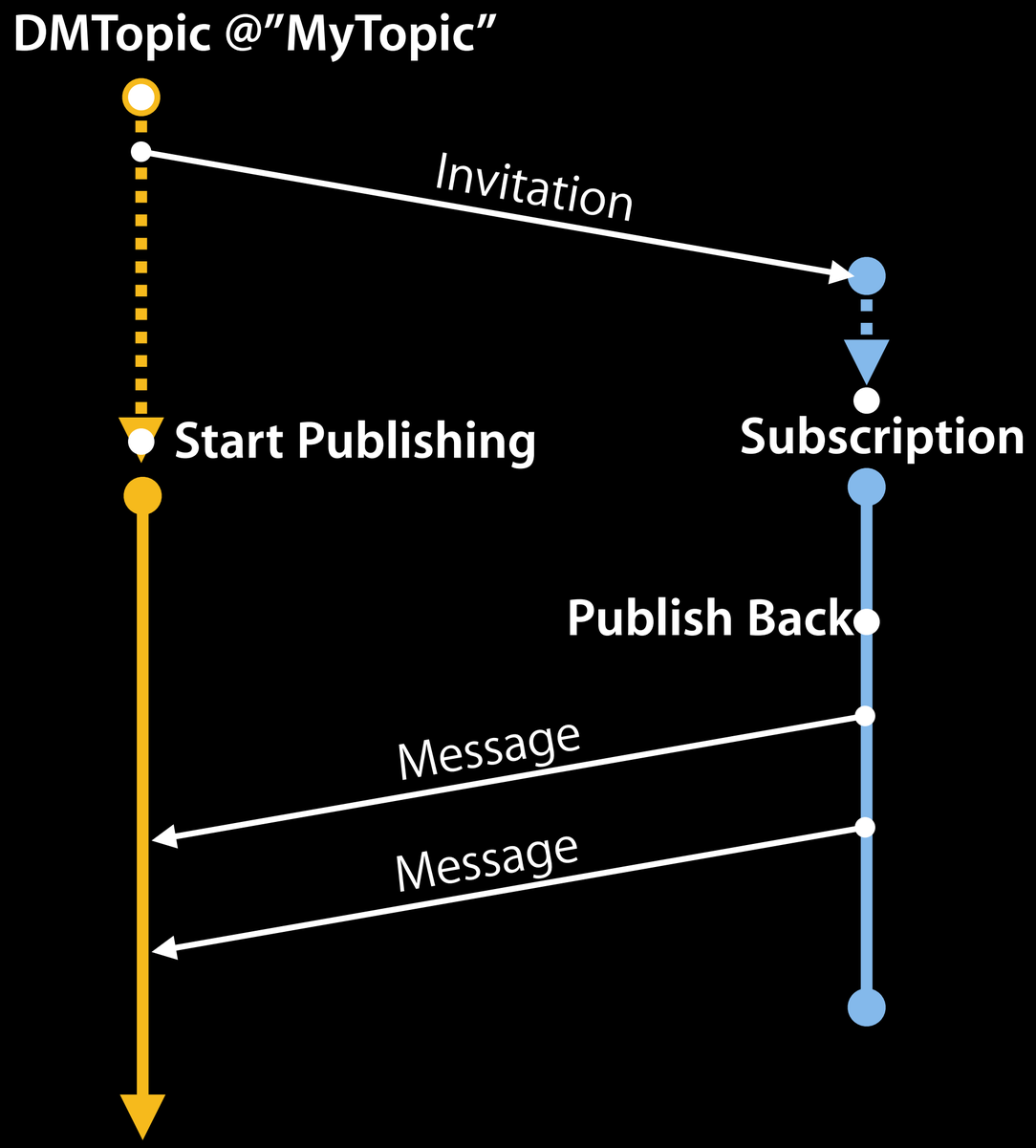
- Accept ones you want

```
if ([transaction isSuccessful]) {
   DMTopic *invitationTopic= [[transaction result] lastObject];//You should look deeper
   if ([aController shouldAcceptInvitationToTopic:invitationTopic]) {
         transaction = [subscriber subscribeToTopic:invitationTopic];
   }
}
```

A Little Bit on Security

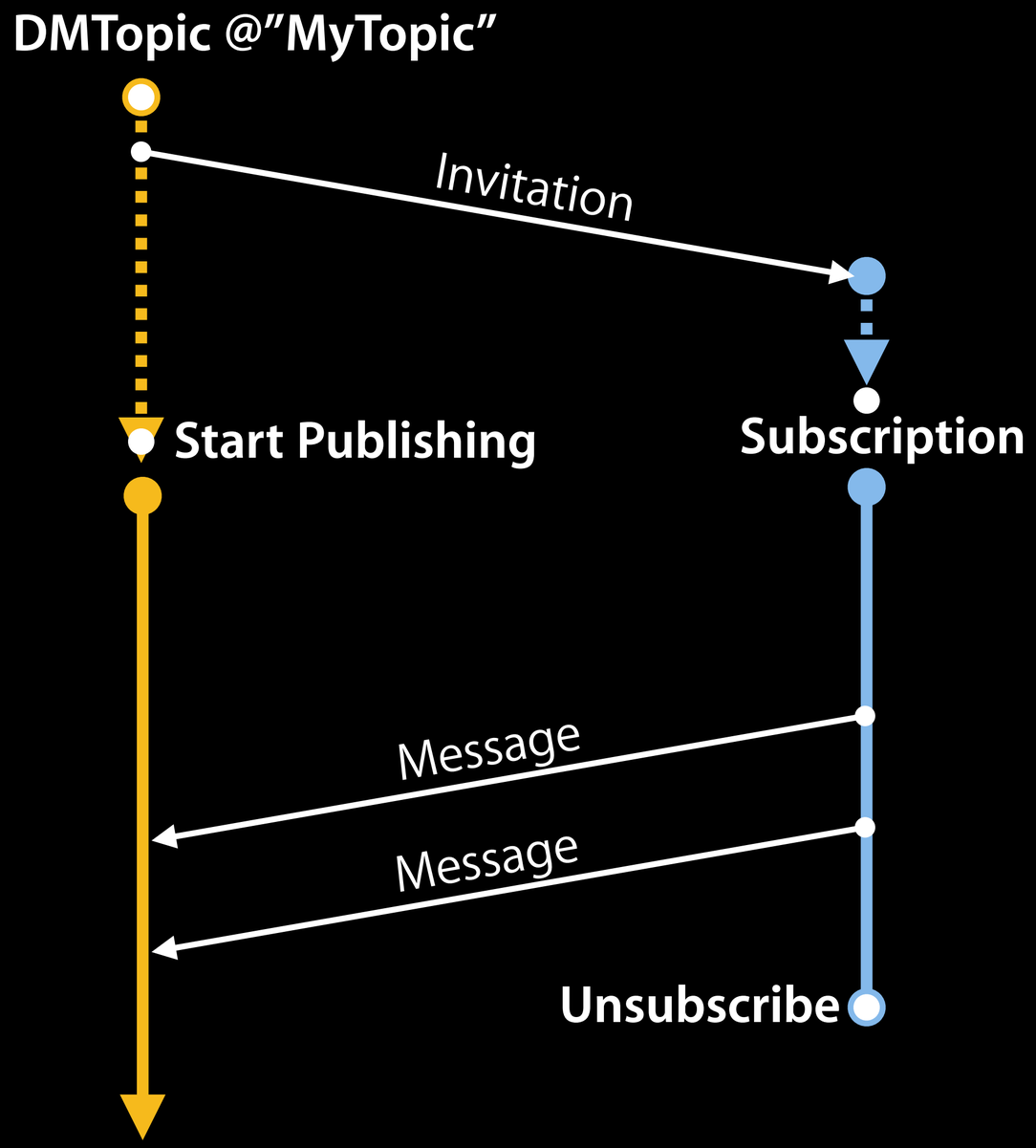
- Can publish at any time
  - All messages are stored
  - Subscribers can “join” anytime





A Little Bit on Security

- Can publish at any time
  - All messages are stored
  - Subscribers can "join" anytime
- "Back" publishing
  - Only sponsor can grant access
  - Sponsor also needs to subscribe



A Little Bit on Security

- Can publish at any time
  - All messages are stored
  - Subscribers can "join" anytime
- "Back" publishing
  - Only sponsor can grant access
  - Sponsor also needs to subscribe
- Anyone can unsubscribe at any time

Very Well Suited To…

- Different applications that need to work together
  - Solution suites
    - Could all be from a single vendor
    - Could be mix and match
  - Components of a workflow
- Moving data from one user of your application to another
  - Picking up and turning in assignments
  - Workflow and approval chains
- Assembling lots of dynamic parts into a whole
  - Multi-user, multi-way collaboration
  - Shared “boards”
- Doing really cool new stuff

Not Well Suited To…

- Large groups
  - Finite number of endpoints
  - Not intended for 1,000:1
- High transaction/message rates
  - kDMServiceBusy
  - Centralized service
  - Invitation and authentication requirements

Not Well Suited To…

- Situations where "messages" are completely transitory
  - All messages are stored centrally
  - There is a maximum capacity based on quota
- Anonymous support (or unknown destinations)
  - Not a P2P system
  - Invitation and authentication requirements

Wrap Up

http://developer.apple.com/wwdc2005/

More Information…

- Talk to the engineering team for .Mac
- Get architectural advice
- Answers for all those burning questions

Lab Session: .Mac SDK 2.0 In Depth	Session 151 Thursday 9:00am

So How Do You Get the Kit?

- Embedded framework you can ship with your application
- Available from http://developer.apple.com/
  - Use a unique creator code for each App
    - Don't use the the samples' @"----" for your application
  - The latest release has
    - The DotMac Kit framework
    - Documentation
    - Sample projects
- Kit 2.0 developer preview 1 (In Universal Binary)

Available July 1, 2005!

The Formula…

Who to Contact

Xavier Legros
Mac OS X Technology Evangelist
xavier@apple.com

Molly Madigan
.Mac Product Marketing, .Mac
mmadigan@apple.com

DotMac Kit Feedback
DotMacKit@mac.com

.Mac Zen
DotMacZen@mac.com

Q&A

.Mac Engineering

http://developer.apple.com/wwdc2005/

WWDC2005

Worldwide Developers Conference